HEWLETT
PACKARD

NS-ARPA/1000
Generation and Initialization

Reference Manual

Software Services and Technology Division
11000 Wolfe Road
Cupertino, CA 95014-9804

Manual Part No. 91790-90030
E0495

Printed in U.S.A. April 1995
Ninth Edition

NOTICE

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

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THE 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.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARs 252.227.7013.

Printing History

The Printing History below identifies the edition of this manual and any updates that are included. Periodically, update packages are distributed which contain replacement pages to be merged into the manual, including an updated copy of this printing history page. Also, the update may contain write-in instructions.

Each reprinting of this manual will incorporate all past updates; however, no new information will be added. Thus, the reprinted copy will be identical in content to prior printings of the same edition with its user-inserted update information. New editions of this manual will contain new information, as well as all updates.

To determine what manual edition and update is compatible with your current software revision code, refer to the Manual Numbering File. (The Manual Numbering File is included with your software. It consists of an “M” followed by a five digit product number.)

First Edition	Feb 1986	Rev. 2608
Update 1	May 1986	Rev. 2626
Update 2	Oct 1986	Rev. 4.1/4010
Second Edition	Aug 1987	Rev. 5.0/5000
Third Edition	Jan 1989	Rev. 5.1/5010
Fourth Edition	Oct 1989	Rev. 5.16/5016
Fifth Edition	May 1990	Rev. 5.2/5020
Sixth Edition	Aug 1991	Rev. 5.24/5240
Seventh Edition	Dec 1992	Rev. 6.0/6000
Eighth Edition	Nov 1993	Rev. 6.1/6100
Ninth Edition	Apr 1995	Rev. 6.2/6200

Preface

Hewlett-Packard Network Services for the HP 1000 (NS-ARPA/1000) provides networking software that allows HP computer systems to communicate with each other.

Audience

NS-ARPA/1000 Generation and Initialization is intended for the Network Manager, whose responsibilities include system planning, and the configuration, generation, and initialization of the HP 1000 systems (also called nodes) in the NS-ARPA network. Besides initially configuring and generating the nodes in the network, the Network Manager also re-generates and re-configures the nodes as is necessary to maintain up-to-date software, and adds nodes, services, and equipment to the network.

Contents

This manual provides the information required to configure an NS-ARPA network, and to generate and initialize the HP 1000 nodes in an NS-ARPA network. This manual also describes how to use Remote I/O Mapping. Remote I/O Mapping is a feature that allows you to map I/O requests from one HP 1000 node to another.

The Network Manager should use this manual in conjunction with *NS-ARPA/1000 Maintenance and Principles of Operation*, part number 91790-90031. This latter manual describes NS-ARPA/1000 maintenance utilities and troubleshooting techniques, and the internal operation of NS-ARPA/1000.

Assumptions

This manual assumes that the Network Manager is familiar with NS-ARPA/1000 and its capabilities, and has read *NS-ARPA/1000 User/Programmer Reference Manual*, part number 91790-90020. The Network Manager should also be familiar with the RTE-A operating system and system generation procedure.

In addition, the Network Manager should be familiar the operating systems of any other machines in the network and any link subsystems used, such as LAN/1000 or X.25/1000.

Organization

Section 1 **Overview**—describes basic network concepts and defines several key terms used in this manual. This section also gives a brief description of the product.

Section 2 **NS-ARPA Installation Summary**—a checklist and brief description of the tasks required to configure, generate, and initialize an NS-ARPA/1000 node.

Section 3 **Network Configuration Planning**—describes how to configure a network and assign items such as node names and network addresses.

Section 4 **Internal Resources**—describes the internal resources required for NS-ARPA/1000, including memory, system resources, and software modules. This section also includes summaries and guidelines for allocating the above resources.

Section 5 **System Generation**—describes the RTE system generation requirements for NS-ARPA/1000.

Section 6 **NS-ARPA/1000 Program and File Installation**—describes how to link and install NS-ARPA/1000 programs and files.

Section 7 **BOOT and WELCOME Files**—describes the tasks to perform from the BOOT command file and system WELCOME file.

Section 8 **Link Initialization**—describes any additional tasks needed in order to use link (such as LAN/1000) subsystems.

Section 9 **NSINIT, NS-ARPA/1000 Initialization**—describes the NS-ARPA/1000 initialization program, NSINIT. NSINIT also shuts down NS-ARPA/1000.

Section 10 **INETD**—describes the Internet Network Services Daemon (INETD) that handles incoming connections for the FTP, TELNET, SMTP (Mail/1000), and Remote Print Spooling services.

Section 11 **NRINIT, Nodal Registry Configuration**—describes the Nodal Registry and the utility used to initialize the Nodal Registry, NRINIT

Section 12 **Additional Initialization Tasks**—describes additional initialization tasks required for TELNET, Remote I/O Mapping, Bisync links (DSLIN), and the DS/1000-IV Compatible Service for transferring files to an HP 3000 TELNET.

Section 13 **Remote VCP (DSVCP) and Forced Cold Loads**—describes the remote Virtual Control Panel utility (DSVCP) and Forced Cold Loads.

Section 14 **Remote I/O Mapping**—describes the Remote I/O Mapping utility.

Appendix A **Configuration Worksheets**—provides blank and completed configuration worksheets, sample NSINIT answer files, and a sample NRINIT configuration file.

Appendix B	**Point-to-Point Network Topologies**—describes and discusses different topologies available for Point-to-Point networks.
Appendix C	**Planning A Memory-Based NS-ARPA System**—includes memory-based system considerations when creating a memory-based NS-ARPA system.
Glossary	Defines NS-ARPA/1000 terms.
Bibliography	Lists other HP networking manuals. To obtain part numbers of these manuals, refer to this section.

Guide to NS-ARPA/1000 Manuals

The following are brief descriptions of the manuals included with the NS-ARPA/1000 product.

91790-90020 NS-ARPA/1000 User/Programmer Reference Manual

Describes the user-level services provided by NS-ARPA/1000. The NS services are network file transfer (NFT), network interprocess communication (NetIPC), and remote program management (RPM). The ARPA services are TELNET and FTP. Because these are interactive and programmatic services, this manual is intended for interactive users as well as programmers. It should also be read by Network Managers before designing an NS-ARPA/1000 network so that they will have a clear understanding of the full implications of various NS-ARPA/1000 functions and features.

91790-90030 NS-ARPA/1000 Generation and Initialization Manual

Describes the tasks required to install, generate, and initialize NS-ARPA/1000. This manual is intended for the Network Manager. Before reading this manual, the Network Manager should read the *NS-ARPA/1000 User/Programmer Reference Manual* to gain an understanding of the NS-ARPA/1000 user-level services. The Network Manager should also be familiar with the RTE-A operating system and system generation procedure.

91790-90031 NS-ARPA/1000 Maintenance and Principles of Operation Manual

Describes the NS-ARPA/1000 network maintenance utilities, troubleshooting techniques, and the internal operation of NS-ARPA/1000. The Network Manager should use this manual in conjunction with the *NS-ARPA/1000 Generation and Initialization Manual*. This manual may also be used by advanced users to troubleshoot their applications.

91790-90040 NS-ARPA/1000 Quick Reference Guide

Lists and briefly describes the interactive and programmatic services described in the *NS-ARPA/1000 User/Programmer Reference Manual* and the *NS-ARPA/1000 DS/1000-IV Compatible Services Reference Manual*. The purpose of this guide is to provide a quick reference for users who are already familiar with the concepts and syntax presented in those two manuals. The *NS-ARPA/1000 Quick Reference Guide* also contains abbreviated syntax for certain programs and utilities described in the *NS-ARPA/1000 Generation and Initialization Manual* and the *NS-ARPA/1000 Maintenance and Principles of Operation Manual*. For your convenience, the *NS-ARPA/1000 Quick Reference Guide* also contains a master index of NS-ARPA/1000 manuals. This is a combined index from the NS-ARPA/1000 manuals to help you find information that may be in more than one manual.

91790-90045 NS-ARPA/1000 Error Message and Recovery Manual

Lists and explains, in tabular form, all of the error codes and messages that can be generated by NS-ARPA/1000. This manual should be consulted by programmers and users who will be writing or maintaining programs for NS-ARPA/1000 systems. Because it contains error messages generated by the NS-ARPA/1000 initialization program NSINIT and other network management programs, it should be consulted by Network Managers.

91790-90050 NS-ARPA/1000 DS/1000-IV Compatible Services Reference Manual

Describes the user-level services provided by the DS/1000-IV backward compatible services. These services are Remote File Access (RFA), DEXEC, REMAT, RMOTE, program-to-program communication (PTOP), utility subroutines, remote I/O mapping, remote system download to memory-based DS/1000-IV nodes only, and remote virtual control panel.

91790-90054 File Server Reference Guide for NS-ARPA/1000 and ARPA/1000

Describes information on using and administering the HP 1000 file server, including runstring parameters, files needed for configuration, troubleshooting guidelines, and error messages.

91790-90060 BSD IPC Reference Manual for NS-ARPA/1000 and ARPA/1000

Describes the 4.3 Berkeley Software Distribution Interprocess Communication (BSD IPC) facility on the HP 1000. BSD IPC is a set of programming development tools originally developed by the University of California at Berkeley (UCB). BSD IPC on the HP 1000 offers a programmatic interface for multi-vendor connectivity to other systems with BSD IPC 4.3.

5958-8523 NS Message Formats Reference Manual

Describes data communication messages and headers passed between computer systems communicating over Distributed System (DS) and Network Services (NS) links.

5958-8563 NS Cross-System NFT Reference Manual

Provides cross-system NFT information. It is a generic manual that is a secondary reference source for programmers and operators who will be using NFT on NS-ARPA/1000, NS3000/V, NS3000/XL, NS/9000, NS for the DEC VAX* computer, and PC (PC NFT on HP OfficeShare Network). Information provided in this manual includes file name and login syntax at all of the systems on which NS NFT is implemented, a brief description of the file systems used by each of these computers, and end-to-end mapping information for each supported source/target configuration.

*DEC and VAX are U.S. registered trademarks of Digital Equipment Corporation.

Caution *If you have purchased the 12079A LAN/1000 Direct Driver Access (DDA) product, please note that HP does not support NS-ARPA/1000 under the following condition:*

- NS-ARPA/1000 used with an altered LAN/1000 interface driver.

With RTE-A and NS-ARPA/1000 software revision codes of 5.0 or later, the 12079A DDA product is supported when NS-ARPA/1000 is also installed. There are restrictions applied to user-written LAN programs. Refer to the *LAN/1000 Link Direct Driver Access Manual* for more information.

If you use an Original Equipment Manufacturer's product that incorporates the 12079A LAN/1000 DDA product, HP may require that the OEM product be certified in order to provide support for NS-ARPA/1000. The purpose of such certification is to ensure that the OEM product does not interfere with the NS-ARPA/1000 product. The certification is billable at time and material rates. For information on HP's certification program, contact your HP representative.

HP does not claim compatibility with non-HP implementations of the Internet Protocol (IP) or the Transmission Control Protocol (TCP).

Conventions Used in this Manual

NOTATION **DESCRIPTION**

`nonitalics` Words in syntax statements that 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 that are in italics denote a parameter that 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:

$\begin{bmatrix}\mathtt{A}\\\mathtt{B}\end{bmatrix}$ 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:

$\left\{\begin{matrix}\mathtt{A}\\\mathtt{B}\\\mathtt{C}\end{matrix}\right\}$ 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 that follows is included. For example:

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

means that the following are allowed:

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

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 that 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)
```

□

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.

Table of Contents

Chapter 1
Overview

Chapter 2
NS-ARPA/1000 Installation Summary

Chapter 3
Network Configuration Planning

Chapter 4
Internal Resources

Chapter 5
System Generation

Chapter 6
NS-ARPA/1000 Program and File Installation

Chapter 7
BOOT and WELCOME Files

Chapter 8
Link Initialization

Chapter 9
NS-ARPA/1000 Initialization

Chapter 10
INETD—Internet Network Services Daemon

Chapter 11
NRINIT, Nodal Registry Configuration

Chapter 12
Additional Initialization Tasks

Chapter 13
Remote VCP (DSVCP) and Forced Cold Loads

Chapter 14
Remote I/O Mapping

Appendix A
Configuration Worksheets

Appendix B
Point-to-Point Network Topologies

Appendix C
Planning A Memory-Based NS-ARPA System

List of Illustrations

Tables

1

Overview

This section provides an overview of NS-ARPA/1000 and defines networking terms used by NS-ARPA/1000. The following subsections are presented:

- Product Description
- NS-ARPA/1000 Terminology
- NS-ARPA/1000 Network Architecture

Product Description

NS-ARPA/1000 provides the following features:

- Layered architecture and implementation of industry-standard data communications protocols.
- Its own memory manager and own hardware mapping set, which allows you to have large networks and reduces the amount of system memory used.
- IEEE 802.3 and Ethernet local area network (LAN) links to HP 1000s, HP 3000s, HP 9000s, and PCs, as well as other non-HP systems.
- Subnetting. An internet addressing scheme that allows you to divide up the host address portion of an internet address. A subnetwork address and a host address are defined within the original host address portion. This extra flexibility allows you to have the same network address for several networks.
- Probe protocol. This protocol enables IEEE 802.3 LAN nodes to automatically determine the names and addresses of other IEEE 802.3 nodes on the LAN and reduces the user's configuration task.
- Address Resolution Protocol (ARP). This protocol translates Internet Protocol (IP) addresses to hardware station addresses for Ethernet links.
- BSD IPC. The 4.3 Berkeley Software Distribution Interprocess Communication (BSD IPC) facility provides programmatic interprocess communication for multi-vendor networking.
- ARPA Services. These services are File Transfer Protocol (FTP) for transferring files and TELNET for remote virtual terminal access.

- NS Common Services. These services are user utilities that allow file transfers and resource sharing between remote systems and programmatic utilities for interprocess communication, remote resource control, remote file access, and program-to-program communication. These services are Network File Transfer (NFT), Network Interprocess Communication (NetIPC), and Remote Process Management (RPM).

- Backward compatibility with DS/1000-IV systems. The DS/1000-IV services DEXEC, PTOP, Remote Database Access, REMAT, Remote I/O Mapping, RFA, and Transparent File Access (part of RTE; also called DS Transparency) are supported over HDLC, IEEE 802.3, and X.25 communication links. DS/1000-IV user programs can run on NS-ARPA/1000 nodes with no modification.

- Remote system download utilities and other features that allow unattended operation of remote DS/1000-IV systems and memory-based systems (HDLC and IEEE 802.3 LAN links).

- Network management utilities to retrieve diagnostic information, and to log and trace network activity.

- The HP 1000 file server allows NFS* client systems to access HP 1000 files.

Hardware Requirements

To use NS-ARPA/1000, you need to install the network interfaces and cabling that provide the communication link to other nodes in the network.

NS-ARPA/1000 requires about 800 pages (1.5 megabytes) of main memory. When determining the total amount of memory required by your system, consider the additional requirements of the RTE-A operating system and any other subsystem that is installed.

In addition, a disk-based NS-ARPA/1000 system must have at least one attached disk drive (using 12.5 megabytes).

The TELNET virtual terminal service is only supported on terminals directly connected or connected by modem to the MUX for A400 computers or to the 12040D Serial MUX card for other A-Series computers. The 12040D MUX must have firmware revision 5.02 or later. All other terminal connections are not supported under TELNET.

Software Requirements

NS-ARPA/1000 requires the RTE-A Virtual Code (VC+) System Enhancement Package and the RTE-A Multiuser system.

NS-ARPA/1000 does not support more than six pages of system common. Of the six pages, one page is used by NS-ARPA/1000. If more than six pages are generated into the system, then some NS-ARPA/1000 programs cannot be linked for the system.

* NFS is a trademark of Sun Microsystems, Inc.

NS-ARPA/1000 Terminology

This section defines data communication terms as they are specifically used by NS-ARPA/1000. These terms are also defined in the Glossary.

Nodes and Networks

A *network* is a group of computer systems or *nodes* connected to each other and with the same network address. They are connected to exchange information and share resources. A network of networks is called an *internetwork* or *internet*.

A *node* is a computer in a network. In this manual, *local node* refers to the node where you are physically located and logged on, and at which you enter commands. The *remote node* refers to a node that may not be physically located where you are, and which you communicate with via data communication.

Nodes are connected in a network by *communication links*. The links can be physical, such as cables, or logical, such as telephone or satellite connections. (The links can also be X.25 connections. For more information on X.25, refer to *DSN/X.25/1000 Reference Manual*, part number 91751-90002.) Typically, all nodes in a network are connected by the same type of link.

Node Names and Addresses

Node Names identify nodes. A node name is a symbolic name associated with an internet address, by which a node can be uniquely identified.

Addresses indicate where nodes are located. NS-ARPA/1000 nodes use numeric addresses called IP addresses.

Node names and IP addresses are described in detail in Section 3, “Network Configuration Planning.”

In Figure 1-1, Network A has five nodes, each represented by a circle. The links joining the nodes are indicated by lines.

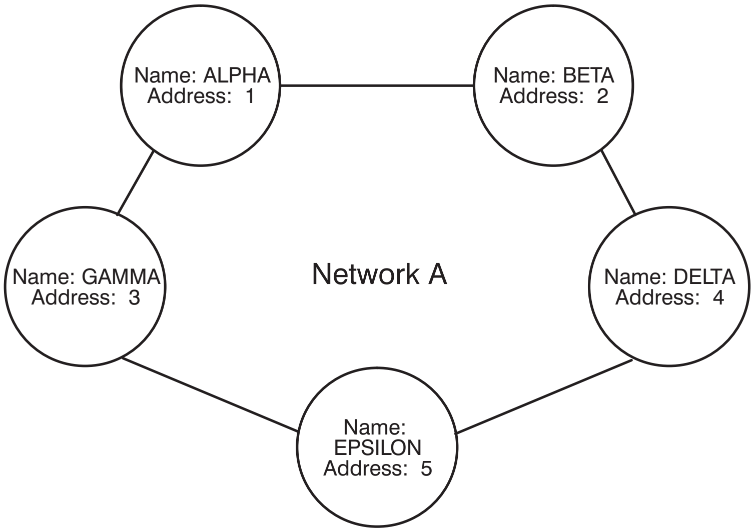


Figure 1-1. Example of a Network, Network A, with 5 Nodes

If you were sitting at and logged on at node ALPHA, node ALPHA would be your *local node*; nodes BETA, GAMMA, DELTA, and EPSILON would be *remote nodes*. If you wanted to send a message from node ALPHA to node BETA, node ALPHA would be the *source node*, and node BETA would be the *destination node*. For messages sent from node BETA to node ALPHA, node BETA would be the source node and node ALPHA would be the destination node.

Routes, Paths, and Protocols

A *route* is the course that a message takes through the network, from the source node to the destination node. A route can pass through intervening nodes.

A *path* is the course that a message takes within a machine, typically through software *protocol handlers*.

A *protocol* is a set of rules for a particular communication task. A protocol handler or protocol module is a piece of software that implements a particular protocol.

Point-to-Point Networks

Network A in Figure 1-1 is a *point-to-point network*; communication travels from one node (point) to another node over the links. NS-ARPA/1000 also supports *broadcast networks*, as explained below.

Store and Forward

In Figure 1-1, there is no direct link between nodes ALPHA and DELTA. To send a message from node ALPHA to node DELTA, node ALPHA could first send the message to node BETA; node BETA would receive the message, determine that it is for node DELTA, and forward the message to node DELTA. Such a transaction is called *store-and-forward*; node BETA is the store-and-forward or intermediate node; node ALPHA is the source node; and node DELTA is the destination node. Messages can be stored and forwarded between several nodes; for example, node BETA could communicate with node GAMMA by first sending the message to node DELTA; node DELTA would forward it to node EPSILON, which would send the message to node GAMMA, the destination node.

Any pair of nodes in Figure 1-1 that does not have direct links between them would have to use a store-and-forward service to route the message from the source node through intermediate nodes to the destination node.

Dynamic Rerouting

Rerouting is the capability to reroute messages around inoperative links or nodes. For example, in Figure 1-1, if the link between nodes ALPHA and BETA failed, node ALPHA could still send a message to BETA by routing the message through nodes GAMMA, EPSILON, and DELTA.

Dynamic Rerouting automatically reroutes messages around inoperative HDLC links or nodes without user intervention. Dynamic Rerouting is an option provided by the *Router/1000* protocol. HP recommends that you use this feature on all links that support it.

Topologies

There are several topologies you can configure with point-to-point networks. Refer to Appendix B for more information.

Broadcast Networks

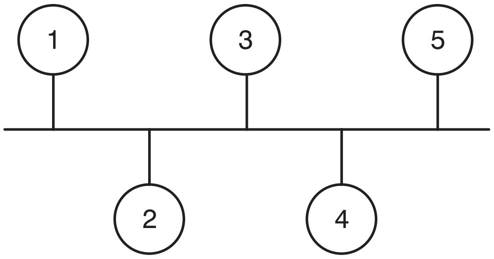


Figure 1-2. Example of a Broadcast Network

Figure 1-2 is an example of a broadcast network. NS-ARPA/1000 supports IEEE 802.3 local area networks (LANs) which are broadcast *bus networks*. In a bus network, the nodes are connected by a linear run of cable. Messages are simultaneously transmitted to every node. Typically, the nodes will process only those messages addressed to them, and ignore all other messages.

Network Boundaries

With NS-ARPA, you can join several networks to form a network of networks, called an *internetwork* or *internet*. The divisions between the networks are called *network boundaries*. In Figure 1-3, the network boundary is indicated by dashes.

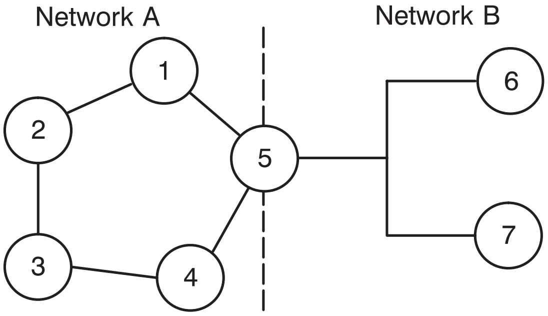


Figure 1-3. Internetwork

Local (Directly Connected) and Remote Networks

Just as there is a distinction between local and remote nodes, there are local and remote networks. The *local network* is the network to which the local node belongs. The local network is also referred to as the *Directly Connected Network* (*DCN*). A *remote network* is any network in the internetwork to which the local node does not belong.

Gateways

Networks are joined together by *gateway nodes*. A gateway node is a node that is a member of two or more networks and allows communication between the networks to which it belongs. Communication between networks is also called *internetwork communication*. For example, in Figure 1-4, Node 11 belongs to Network C and to Network D. Node 11 serves as a gateway between Networks C and D. Internetwork protocols determine how data communication messages are routed through networks. NS-ARPA/1000's *Internet Protocol (IP)* is based on the DARPA (Defense Advanced Research Projects Agency) standard.

If Node 7 in Network C wants to send a message to Node 14 in Network D, Node 7's message would be sent to Node 11. Node 11 would then send the message to Node 14. All data communications traffic from Network C to Network D must go through Node 11, the gateway to Network D.

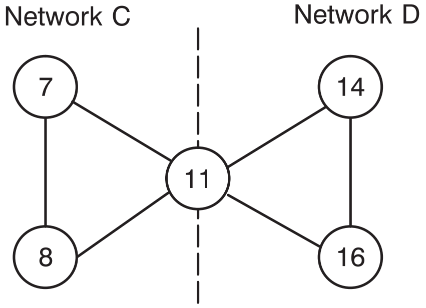


Figure 1-4. Gateway (Node 11)

Routing Information

All nodes in the same network have routing information for all the nodes in that network. Information used to reach other nodes in a network is called network routing information. Gateways have routing information to reach other gateways, and therefore, other networks. Information used to reach other networks is called internetwork routing information.

In Figure 1-5, Node 1 in Network A does not have detailed network routing information about Network C, except that it must send all messages destined for Network C to Node 5, the gateway to Network B. Node 5 has internetwork routing information, and is responsible for routing the messages to Node 9, the gateway to Network C. Node 9 is then responsible for network routing within Network C.

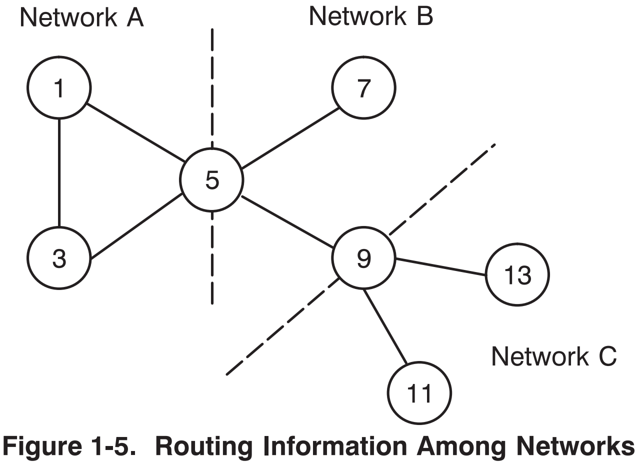


Figure 1-5. Routing Information Among Networks

NS-ARPA/1000 Network Architecture

A *network architecture* is a structured, modular design for networks. The International Standards Organization (ISO) has developed a network architecture model called *Open Systems Interconnection (OSI)*. HP used this model to guide NS-ARPA development.

In the OSI model, transmission and communication tasks are assigned to logically distinct modules called *layers*. Each layer communicates with the layer directly above it and below it, and through the layers below it to its peer in the remote computer. A layer does not need to know how the other layers perform their tasks, but needs to know how to use the services of the layer below it and how to provide services to the layer above it.

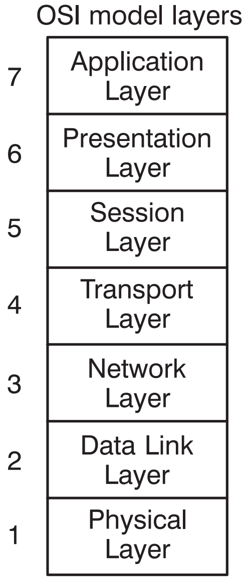


Figure 1-6. OSI Model

The OSI model defines seven layers with the following tasks:

- *Layer 7 (Application)* is the user interface to remote services.
- *Layer 6 (Presentation)* manipulates user data, such as in text compression and encryption.
- *Layer 5 (Session)* negotiates connection establishment at remote nodes.
- *Layer 4 (Transport)* is responsible for *end-to-end* data integrity. End-to-end indicates that Layer 4 communicates with its peer only at the *source* and *destination* nodes, not at intermediate nodes. Layers 5-7 also provide end-to-end services, while Layers 1-3 are responsible for data integrity between each node.

  Layer 4 protocols can provide the following services to upper layers: in-order message delivery, retransmission of lost messages, suppression of duplicate messages, and flow control.
- *Layer 3 (Network)* determines the routes messages take to get from one node to another. This layer can be split into two sub-layers: *3i* and *3s*. Layer 3i handles internetwork routing and 3s handles subnetwork or *intranetwork* routing, which is routing *within* a network.
- *Layer 2 (Data Link)* checks for and corrects transmission errors over the physical link.

- *Layer 1 (Physical)* transmits the electrical signals over the link.

Note that Layers 1 through 4 are also collectively referred to as the *transports*.

Figure 1-7 on the following page describes NS-ARPA/1000 using the OSI model as a reference.

Application Layer

The NS-ARPA/1000 Application Layer (Layer 7) services are divided into five groups: BSD IPC, ARPA Services, NS Common Services, DS/1000-IV Compatible Services (RTE-RTE), and DS/1000-IV Compatible Services (RTE-MPE). The following description of user services is an overview only. For more information on BSD IPC, refer to the *BSD IPC Reference Manual for NS-ARPA/1000 and ARPA/1000*, part number 91790-90060. For the NS and ARPA services, refer to the *NS-ARPA/1000 User/Programmer Reference Manual,* part number 91790-90020. For DS/1000-IV services, refer to the *DS/1000-IV Services User/Programmer Reference Manual,* part number 91790-90050.

BSD IPC

The 4.3 Berkeley Software Distribution Interprocess Communication (BSD IPC) facility provides programmatic interprocess communication for multi-vendor networking.

ARPA Services

The ARPA services are as follows:

- *TELNET* allows you to have a virtual terminal connection to another computer in your network.
- *File Transfer Protocol (FTP)* allows you to copy files from one node to another interactively or programmatically.
- *Time Server Protocol* allows you to synchronize clocks with other systems in your network.

TELNET and FTP are available on other systems which support ARPA services.

File Server

The HP 1000 file server allows NFS client systems to access HP 1000 files.

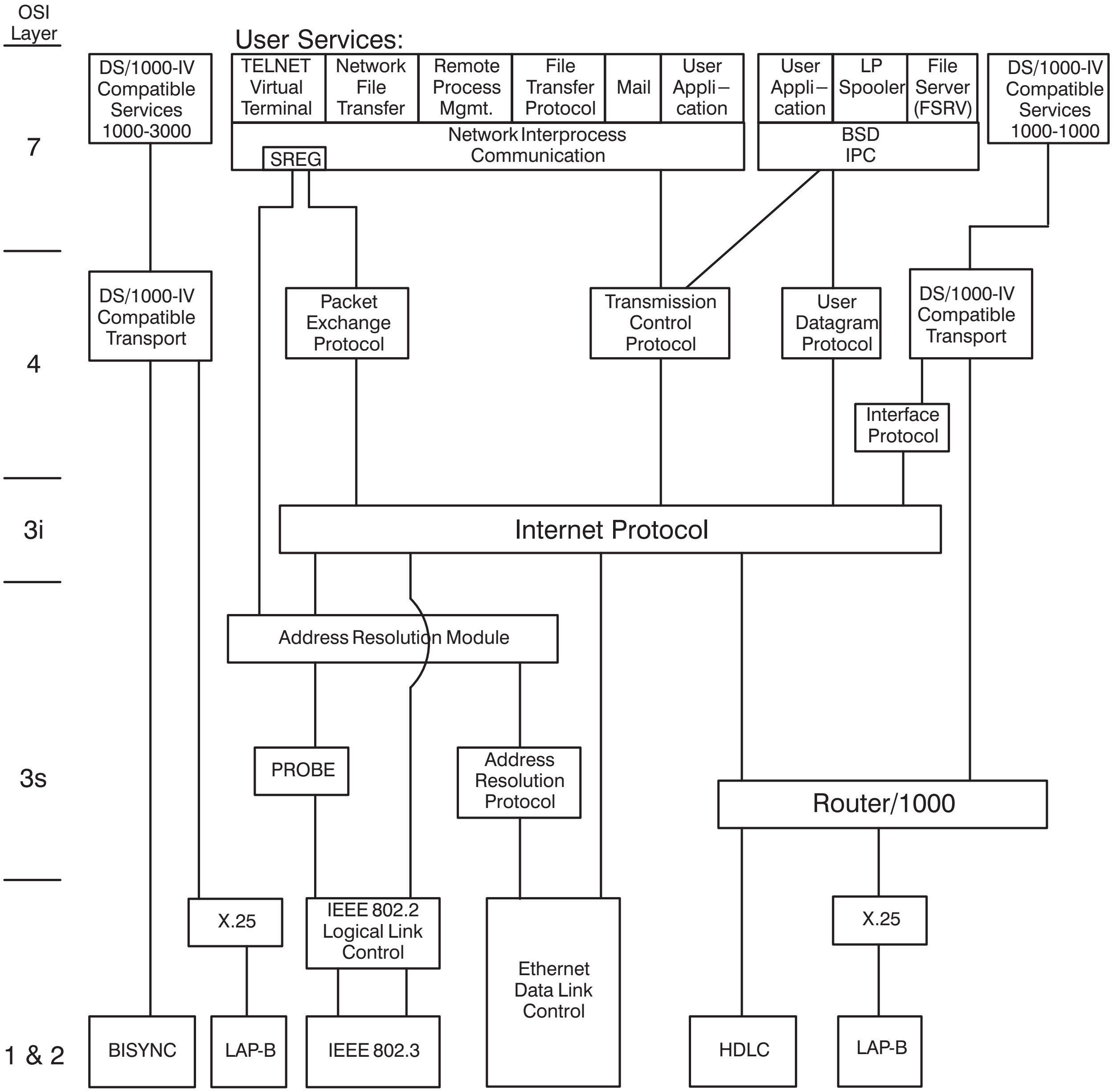


Figure 1-7. NS-ARPA/1000 Architecture

NS Common Services

The NS Common Services are as follows:

- *Network File Transfer (NFT)* allows you to copy files from one node to another interactively or programmatically.

- *Network Interprocess Communication (NetIPC)* allows independent processes running concurrently on different nodes to exchange information in a peer-to-peer manner. NetIPC processes communicate with each other via sockets. NetIPC uses the *Socket Registry (SR)* service. SR allows NetIPC to “look up” a socket by name at a remote node. SR then returns an identifier for the logical connection, which is used to communicate with the remote socket. Socket Registry is not directly accessible by the user.

- *Remote Process Management (RPM)* allows you to schedule or terminate programs located at the same or different HP 1000 node in your network.

NFT and NetIPC are available on other HP systems for cross-system use: HP 9000 Series 300 and 800, HP 3000, and PC. NFT is also supported to the DEC VAX computer with the “NS for the DEC VAX Computer” product.

DS/1000-IV Compatible Services (RTE-RTE)

The following DS/1000-IV Compatible Services (RTE-RTE) can be used to communicate with NS-ARPA/1000 nodes and for backward compatibility to DS/1000-IV nodes:

- *Distributed Executive (DEXEC)* allows you to control I/O devices located at remote HP 1000 nodes. DEXEC calls are remote calls equivalent to local RTE EXEC calls.

- *Program-to-Program Communication (PTOP)* enables a “master” program on one node to send information to and control the execution of a “slave” program on a remote HP 1000 node.

- *REMAT* allows you to send RTE operator commands and special REMAT commands to a remote HP 1000.

- *Remote Database Access (RDBA)* allows you to access an IMAGE database at a remote HP 1000. (Refer to the appropriate IMAGE manual for more information.)

- *Remote File Access (RFA)* enables you to perform I/O operations to files located at remote nodes.

- *Transparent File Access (TRFAS)* allows you to access remote files. TRFAS is part of the RTE operating system and also is called DS File Transparency. Its usage is documented in the RTE manuals.

- *Utility Subroutines* enable you to perform tasks such as downloading absolute or memory-image program files to memory-based nodes and programmatic remote logons to RTE-6/VM systems.

- *Remote I/O Mapping* maps I/O requests from one node to another, allowing resource sharing. Refer to the “Remote I/O Mapping” section of this manual for more information.

- *Remote System Download* allows you to download an operating system file to a remote HP 1000. Refer to the "Remote VCP (DSVCP) and Forced Cold Loads" section of this manual for more information. This feature is available only over HDLC links configured with Router/1000 link interfaces.
- *Remote Virtual Control Panel (VCP)* allows you to operate the VCP of a remote node. It is useful for applications requiring a remote, terminal-less node (such as harsh operating environments). Refer to the "Remote VCP (DSVCP) and Forced Cold Loads" section of this manual for more information. This feature is available only over HDLC links configured with Router/1000 link interfaces.

DS/1000-IV Compatible Services (RTE-MPE)

The following DS/1000-IV Compatible Services (RTE-MPE) can be used to communicate with MPE-V NS/3000 nodes that have DS/3000 and for backward compatibility to DS/3000 nodes.

- *Program-to-Program Communication (PTOP)* enables a "master" program on one node to send information to and control the execution of a "slave" program on a remote HP 3000 node.
- *Remote File Access (RFA)* enables you to perform I/O operations to files and peripherals located at remote nodes.
- *RMOTE* creates an interactive session on a remote HP 3000, making your terminal appear to be directly connected to the HP 3000.
- *Utility Subroutines* enable you to perform tasks such as programmatic remote logons.

You can also use most of the above services for strictly local applications. For example, you can use NetIPC for communication between two processes running on the same node.

Session Layer

NS-ARPA has no Session Layer; however features in the Network Interprocess Communication (NetIPC) service and the DS/1000-IV Transport provide some of the functionality of such a layer.

Presentation Layer

NS-ARPA has no Presentation Layer; however the Network File Transfer service converts some file attributes.

Transport Layer

At the Transport Layer (Layer 4), NS-ARPA/1000 implements the*Transmission Control Protocol (TCP)* and the *User Datagram Protocol (UDP)* based on the DARPA standards, and the *Packet Exchange Protocol (PXP)*. PXP is an HP protocol that provides low-overhead *datagram* service.

Transmission Control Protocol (TCP)

TCP provides non-duplicated, in-sequence data delivery. TCP is a stream-based (rather than message-based) protocol. TCP accepts arbitrarily long data buffers, segments them into packets and sends each packet separately. TCP keeps track of the bytes sent and retransmits them if they are not acknowledged within a timeout interval. TCP at the receiving node reassembles the packets, so that they are delivered to the user (NetIPC, FTP, and TELNET) in order (in-sequence delivery).

TCP is a connection-based protocol. When the TCPs at two nodes want to communicate, they establish a logical communication channel called a connection. Establishing a TCP connection requires overhead because each node must allocate buffers and other resources to support the connection, and because the TCPs must perform a connection handshake before any data is sent. TCP also provides flow control; the amount of data sent can be controlled so that the sender does not overload the receiver.

All NS Common Services and the ARPA Services, FTP and TELNET, use TCP.

User Datagram Protocol (UDP)

User Datagram Protocol (UDP) provides unreliable, connectionless data delivery. Messages can be duplicated, out of sequence, or missing. Messages retain their record boundaries and are sent as individually addressed packets. UDP is used by the HP 1000 file server.

Packet Exchange Protocol (PXP)

Packet Exchange Protocol (PXP) is a low-overhead request/reply message-based protocol that is suited for querying data sources. Since PXP does not establish connections, there is no overhead for connection establishment. However, because the amount of overhead is the same for each transaction, subsequent transactions cannot take advantage of an established connection. PXP suppresses duplicate replies to a request but does not suppress responses to duplicate requests. PXP retransmits messages that are not acknowledged within a timeout interval.

Socket Registry is the only service that uses PXP; it is not user accessible.

DS/1000-IV Compatible Transport

DS/1000-IV Compatible Services (including Transparent File Access) use the *DS/1000-IV Compatible Transport (RTE-RTE and RTE-MPE)*. The DS/1000-IV Compatible Transport (RTE-RTE) uses Message Accounting *(MA)*. MA is a datagram-oriented protocol that retransmits lost messages and suppresses duplicate messages. *IFP*, the Interface Protocol, is the interface for backward compatible services to Network Layer, IP.

Network Layer

At the Network Layer (Layer 3), NS-ARPA/1000 has Internet (3i) and Subnet or Intranetwork (3s) protocols. *Internetwork* protocols route data *between* networks, and *intranetwork* protocols route data *within* networks.

Internet Layer

At the Internet Layer (3i), NS-ARPA implements the *Internetwork Protocol (IP)* based on the DARPA standard. IP is primarily used to route messages between networks via gateways. It provides gateway-to-gateway routing, store-and-forward service between gateways, and message fragmentation and reassembly between source and destination networks. NS-ARPA nodes have IP addresses which consist of a network address and a node address. IP addressing also includes subnetting in which the node address can be divided into a subnet number and a node number. IP addressing and subnetting are further discussed in “IP Address” of Section 3, “Network Configuration Planning.”

Subnet or Intranet Layer

At the Subnet or Intranet Layer (3s), NS-ARPA offers the *Router/1000*, IEEE 802.3, Ethernet, and X.25 protocols. Router/1000 provides *store-and-forward* and *Dynamic Rerouting* services for messages sent over HDLC links. The store-and-forward service allows you to access non-adjacent nodes; messages are automatically forwarded through intermediate nodes. The optional Dynamic Rerouting service automatically reroutes messages around inoperative links. In addition, the IEEE 802.3, Ethernet, and X.25 protocols define some layer 3s functions for messages sent over those links. With IEEE 802.3 and Ethernet networks, messages are transmitted to every node in the network and a node accepts only messages addressed to it; this makes store-and-forward and rerouting services unnecessary. X.25 networks also define routing and store-and-forward features within Packet-Switching Networks.

Address Resolution Module (ARM)

ARM, the Address Resolution Module, is a central “switch” that coordinates the interactions between the following two:

- any address resolution requestor
- the particular address resolution provider

ARM’s clients are socket registry and IP. ARM’s resolver protocols are Probe and ARP. ARM is not directly accessible to users. It is for HP internal use only.

Probe

Probe is an HP protocol that is used to translate user-defined node names into Internet Protocol (IP) addresses. Probe multicasts the name of a node to all of the nodes in the local network. The node that is associated with the node name being broadcast identifies itself by replying to Probe with its IP addresses and protocols supported. Probe also translates Internet Protocol addresses to hardware addresses, also called station addresses or link-level addresses. Probe has very low overhead. Probe broadcasts its inquiries on IEEE 802.3 links only.

Probe is not directly accessible to users. It is for HP internal use only.

Address Resolution Protocol (ARP)

ARP, the Address Resolution Protocol, provides similar functionality to Probe. ARP translates Internet Protocol (IP) addresses to hardware station addresses. However, ARP does not translate user-defined names into machine-readable addresses. ARP broadcasts its inquiries on Ethernet links only.

Like Probe, ARP is not directly accessible to users. It is for HP internal use only.

Data Link and Physical Layers

The Data Link and Physical Layers (Layers 2 and 1) are implemented on interface cards. NS-ARPA/1000 supports the following Data Link (Layer 2) protocols: HDLC, Bisync, IEEE 802.3 (the IEEE 802.3 protocol encompasses the IEEE 802.2 Layer 2 protocol), Ethernet LAN, and X.25 (X.25 encompasses the LAP-B protocol).

Communication Links

Communication links are characterized by the services and physical topologies they support, by line distances and speed, and by hardware costs. This subsection briefly describes each type of communication link. The line speeds and distances for each link are summarized below:

- *IEEE 802.3 or Ethernet LAN*—full duplex, 10 mbps, 500 meters per segment
- *HDLC*—full duplex, up to 230 kbps
- *BISYNC*—half duplex, up to 56 kbps
- *X.25*—full duplex, up to 56 kbps, limited by the data transmission rate of the Public Data Network used

IEEE 802.3 Local Area Network Link

IEEE 802.3 links join HP 1000 systems with other HP computers.

All NS-ARPA services are supported on IEEE 802.3 links between HP 1000s except for the following services:

- DS/1000-IV Compatible Services (RTE-MPE) runs over Bisync and X.25 links only.
- Remote Virtual Control Panel operation (DSVCP) and remote system download (PROGL, FCL66) runs over HDLC only. Remote VCP and system download over a LAN is available from RTE-A software.

IEEE 802.3 links are fast and require less hardware than point-to-point links for multi-node networks. A LAN is formed when such joining occurs over a relatively small geographical area.

IEEE 802.3 networks are broadcast bus networks. Messages are transmitted to every node, and if one node fails, the other nodes can still communicate. For more information on bus topologies, refer to the *LAN Cable and Accessories Installation Manual*.

IEEE 802.3 LANs require only one interface card at each node, regardless of the number of nodes on the LAN. IEEE 802.3 networks provide flexible node addition and deletion. To add an IEEE 802.3 node, you need to install one card at the new node.

Ethernet Local Area Network Link

Ethernet is a de-facto standard link-level protocol. It was developed before IEEE 802.3 was defined. IEEE 802.3 evolved from Ethernet. Like IEEE 802.3, Ethernet also defines a baseband, coaxial, bus media, and the Media Access Method CSMA/CD. IEEE 802.3 and Ethernet nodes can coexist on the same cable, but cannot communicate with each other.

HDLC

You can use HDLC links only for HP 1000−HP 1000 connections. HDLC links are the only links that support remote Virtual Control Panel operation (DSVCP) and DS/1000-IV Compatible Services (RTE-RTE). Therefore, HDLC links are mostly used for NS-ARPA/1000 to DS/1000-IV connectivity.

HDLC connections require two interface cards, one at each node. To add another HDLC connection, you must have two additional cards, one at each node, and an additional cable. HDLC links can be hardwired or modem connections.

Bisync

You can use Bisync links only for NS-ARPA/1000−NS/3000 and NS-ARPA/1000−DS/3000 connections. Only DS/1000-IV Compatible Services (RTE-MPE) are supported over Bisync links.

Bisync links have no store-and-forward or rerouting capabilities. HP recommends that you configure HP 3000 nodes connected via Bisync links as leaf nodes (a node not used for store-and-forward traffic).

Bisync connections require two interface cards, one at each node. To add another Bisync connection, you must have two additional cards, one at each node, and an additional cable. Bisync links can be hardwired or modem connections.

X.25

X.25 links provide connections to Packet Switching Networks (PSNs), also known as Value Added Networks (VANs). X.25 links are useful for long-distance communication, and can be more economical than leased lines in some applications.

X.25 links to HP 3000s have no store-and-forward capabilities. HP recommends that you configure HP 3000 nodes connected via X.25 links as leaf nodes.

X.25 links require X.25 software (HP part number 91751A/R). For more information, refer to *DSN/X.25/1000 Reference Manual*.

2

NS-ARPA/1000 Installation Summary

This section provides an installation overview for installing NS-ARPA/1000. This is only an overview. If you need detailed information on the steps listed here, refer to the respective sections in this manual or in the RTE-A manuals. The steps are divided into the following groups:

- Plan your network.
- Install the software.
- Regenerate the system.
- Install NS-ARPA/1000 programs.
- Start up NS-ARPA/1000 networking.
- Verify your system.
- Shut down your system.
- Modify the BOOT and WELCOME files.

The steps are presented for a user installing NS-ARPA/1000 for the *first* time. After you are comfortable with system generations with NS-ARPA/1000, you will know which steps can be shortened, skipped, or modified. If you are receiving an update of NS-ARPA/1000 through support subscription services, you will receive a *Communicator/1000*. There are updating procedures in the *Communicator/1000* for each release.

An installation overview is provided on the NS-ARPA/1000 product tape in the file `/NS1000/A91790_INSTALL.TXT`. The file `/NS1000/A91790_EZ_INSTAL.TXT` describes how to install NS-ARPA/1000 with a LAN only link and no DS/1000-IV Compatible Services.

Where appropriate, sections in this manual and example files are referenced. The generation, configuration, and initialization example files are located in `/NS1000/EXAMPLES`. These example files are included with NS-ARPA/1000 as learning tools only; you should use the information in this manual to design configuration files that meet your needs.

If your network includes LAN links to HP 3000, HP 9000, PC, or other systems, refer to that computer's networking documentation for installation information.

You must keep up-to-date copies of the system generation answer file, NSINIT answer file, installation, BOOT, and WELCOME files. Should you have problems with your system, HP will require these files.

Note Do not install NS-ARPA/1000 on a system with ARPA/1000 (product 98170A/R) already installed.

Plan Your Network

Before you can bring up your NS-ARPA/1000 system, you must plan and determine how to configure your node in the network.

1. Plan your network (or internet) and your NS-ARPA/1000 node configuration. Configuration includes network boundary definitions, and name and address assignments.

   Section Reference: Section 3, "Network Configuration Planning."

2. Create a network worksheet for each node in the network. A blank network worksheet and example worksheets are provided in Appendix A, "Configuration Worksheets."

   Section Reference: Appendix A, "Configuration Worksheets."

Install the Software

When you are ready to install the NS-ARPA/1000 software from the product tape, make sure you have a good working system.

1. Install and verify the necessary hardware.

   Reference: the appropriate hardware reference manual.

2. Install and verify that your RTE-A system runs properly.

   Reference: the RTE-A manuals.

3. Create the `/NS1000` directory on an LU that has at least 21,000 blocks of disk space. Otherwise, the RTE `TF` utility automatically creates `/NS1000` on the first available LU.

   ```
   CI> CRDIR /NS1000 lu
   ```

An additional 17,430 blocks of disk space is needed to install NS-ARPA/1000 programs, libraries, and other files into the target directories. Refer to the subsection, "Install NS-ARPA/1000 Programs," in this section and to the subsection, "Disk Space," in Section 4.

4. Copy the NS-ARPA/1000 files from the product tape to your system, using the RTE `TF` utility.

   Reference: *RTE-A Backup and Disk Formatting Utilites Manual*, part number 92077-90249.

   `CI> TF,CO,lu` where *lu* is the LU of the tape device.

The file /NS1000/MISC/A91790.SNF lists the /NS1000 subdirectories created and describes all of the NS-ARPA/1000 files.

Example Files:

```
/NS1000/A91790_INSTALL.TXT
/NS1000/A91790_EZ_INSTAL.TXT
/NS1000/CMD/INSTALL_NS1000.CMD
```

Regenerate the System

There are certain modules required in a system generation for NS-ARPA/1000. Prepare to regenerate your system.

1. Back up your current system, and verify your backup copy.
2. Look at the example answer file, /NS1000/EXAMPLES/#ANSNS . Determine how you want to regenerate your system.
3. Determine the *internal resources* required for NS-ARPA. This includes memory, system resources, and software modules.

   Section Reference: Section 4, "Internal Resources."
4. Determine the number of TELNET pseudo terminal LUs and modem LUs needed. If more than eight total LUs are desired, you must reconfigure the HPMDM_TABLE table to increase the number of LUs before system generation.

   Section Reference: Section 5, "System Generation" and Section 12, "Additional Initialization Tasks."
5. Modify your RTE system generation answer file for your link(s) and NS-ARPA/1000.
6. Regenerate your system.

   ```
   CI> RTAGN your_answer_file
   ```

   Section Reference: Section 5, "System Generation."
   Example File: /NS1000/EXAMPLES/#ANSNS
7. Create the /SCRATCH directory. FTP uses the /SCRATCH directory for temporary files. The directory must not be write-protected.

   ```
   CI> CRDIR /SCRATCH lu
   ```
8. Boot your new system.

Install NS-ARPA/1000 Programs

Once you have regenerated the system, NS-ARPA/1000 programs must be linked and restored.

1. Set your working directory to `/NS1000`.

   ```
   CI> WD /NS1000
   ```

2. Link the NS-ARPA programs and copy the files using the command file, `/NS1000/CMD/INSTALL_NS.CMD`. The NS-ARPA/1000 programs must be linked with the snap file of your target NS-ARPA/1000 system.

   Section Reference: Section 6, "NS-ARPA/1000 Program and File Installation."
   Example File: `/NS1000/CMD/INSTALL_NS1000.CMD`

   ```
   CI> TR CMD/INSTALL_NS1000.CMD [Q]
   ```

   where specifying `Q` causes the "quiet" mode to be used and default values used for directories, snap files, and libraries. If `Q` is not specified, the `INSTALL_NS1000.CMD` file prompts you for the following information:

   1. destination directories for NS-ARPA/1000 programs and files
   2. directory of the RTE-A relocatable files to link certain programs
   3. snap file to link NS-ARPA/1000 programs
   4. usage of DS/3000 libraries and programs to build user libraries.

Start Up NS-ARPA/1000 Networking

1. Initialize the links (LAN/1000, X.25/1000, etc.).

2. Restore the NS-ARPA programs with RTE `RP` commands. Refer to `/NS1000/EXAMPLES/NSSTART.CMD`. Review the `RP` commands and delete the commands for NS-ARPA/1000 programs you will not use. The `RP` commands for programs related to DS/1000-to-DS/3000 services, DSVCP (remote VCP), and PROGL (remote system download) are commented out with asterisks (`*`). Remove the asterisks to use those `RP` commands.

   The commands in `/NS1000/EXAMPLES/NSSTART.CMD` can be added to your WELCOME file for future reboots. Refer to the subsection, "Modify the BOOT and WELCOME Files."

3. Initialize your node by running NSINIT. Initialize the Nodal Registry by running NRINIT. The `/NS1000/EXAMPLES/NSSTART.CMD` command file also runs NSINIT and NRINIT.

   Create the NS-ARPA/1000 initialization answer file with NSINIT. The contents of the initialization answer file are based on your network map and network worksheets. Example initialization answer files are listed in Appendix A. NSINIT prompts you for information about your node in a "dialogue."

```
CI> RU NSINIT
```

Section Reference: Section 9, "NSINIT, NS-ARPA/1000 Initialization."

If you are satisfied with the dialogue with NSINIT, you are ready to initialize. You can answer `YES` to the start up question in NSINIT.

You can also answer `NO` to the start up question and use the output file as an answer file later.

Running NSINIT with an answer file can be done at boot up in the WELCOME file. Then NS-ARPA networking can be initialized upon boot up.

Example Files:

```
/NS1000/EXAMPLES/NODE1_LAN.NSIN
/NS1000/EXAMPLES/NODE2_LAN.NSIN
/NS1000/EXAMPLES/NODE1_RTR.NSIN
/NS1000/EXAMPLES/NODE2_RTR.NSIN
```

4. (Optional) Create (with a text editor) a file of node entries. This file is used by NRINIT to initialize the Nodal Registry.

   You do not have to initialize the Nodal Registry if you are communicating only with HP systems on the local network.

   You can also run NRINIT from the WELCOME file to initialize the Nodal Registry upon boot up.

   Section Reference: Section 11, "NRINIT, Nodal Registry Configuration."
   Example File: `/NS1000/EXAMPLES/ALL_NODES.NRIN`

5. Initialize the following: TELNET pseudo terminal LUs, Remote I/O Mapping, and Bisync links to HP 3000 computers.

   Section Reference: Section 12, "Additional Initialization Tasks."

6. If you want to use the RMOTE `MO` command, you must install the slave program, `COPY3K.PUB.SYS`, on the HP 3000.

   Section Reference: Section 12, "Additional Initialization Tasks."

7. If you want to use the HP 1000 file server, run FSRV.

   Manual Reference: *File Server Reference Guide for NS-ARPA/1000 and ARPA/1000*, part number 91790-90054.

Verify Your System

Once you have configured and initialized NS-ARPA/1000, verify the configuration of the ARPA, NS Common, and DS/1000-IV Compatible Services, such as TELNET, NFT (`DSCOPY`), REMAT, and RMOTE. Also verify the appropriate services at each node in the internet. Try using the following commands to quickly verify your system:

- NSINF: `A` (print local node information) and `C` (print configured resources) commands
- PING to a remote node for loopback testing
- TELNET to logon to a remote node
- FTP to copy a file to a remote ARPA or NS-ARPA/1000 node
- DSCOPY (Network File Transfer) to copy a file to a remote NS node
- REMAT to a remote DS/1000-IV node
- RMOTE to a remote HP 3000 node

Shut Down Your System

Sometimes, you may need to shut down NS-ARPA. There are two steps to shut down NS-ARPA.

1. Run NSINIT to shut down networking on the system. Enter the shut down response.

   ```
   CI> RU NSINIT
   ```

2. Remove NS-ARPA/1000 programs using RTE `OF` commands. Refer to the command file `/NS1000/CMD/OF_NS.CMD` as an example and change it according to your system. You must shut down NS-ARPA with NSINIT before running this command file.

   ```
   CI> TR /NS1000/CMD/OF_NS.CMD
   ```

Modify the BOOT and WELCOME Files

To initialize networking at boot up, the steps in the "Start Up NS-ARPA/1000 Networking" section can be done in the BOOT and WELCOME files.

Section Reference: Section 7, "BOOT and WELCOME Files."
Example Files: /NS1000/EXAMPLES/NSSTART.CMD and /NS1000/CMD/NSSTART_EZ.CMD

Modify the BOOT file as described in the steps below.

1. Restore NS-ARPA/1000 programs (RTE RP command). The RP commands can also be done in the WELCOME file instead.

2. (Optional) Reserve a partition (RTE RV command) for the NS-ARPA SHEMA #DSAM.

   ```
   RV,151
   ```

   Be sure to re-link MMINIT to assign #DSAM to this reserved partition.

   ```
   CI> LINK MMINIT.RUN
          LK
          EM size
          SH #DSAM partition_num
          EN
   ```

3. (Optional) Assign programs to reserved partitions (RTE AS commands).

Modify the WELCOME file as described in the steps below.

4. Restore programs that are not RPed in the BOOT file.

5. Make sure the links (LAN/1000, X.25/1000, etc.) are initialized.

   Section Reference: Section 8, "Link Initialization."

6. Run NSINIT to initialize NS-ARPA/1000.

   Section Reference: Section 9, "NSINIT, NS-ARPA/1000 Initialization."

7. Run NRINIT to initialize the Nodal Registry.

   Section Reference: Section 11, "NRINIT, Nodal Registry Configuration."

8. Initialize the following: TELNET pseudo terminal LUs, Remote I/O Mapping, and Bisync links to HP 3000 computers.

   Section Reference: Section 12, "Additional Initialization Tasks."

9. If you want to use the HP 1000 file server, run FSRV.

   Manual Reference: *File Server Reference Guide for NS-ARPA/1000 and ARPA/1000*, part number 91790-90054.

Reboot your system to execute the revised BOOT and WELCOME files.

3

Network Configuration Planning

Determining Your Network Configuration

Before initializing the nodes in your network or internet with NSINIT, you must determine your network or internet configuration. Configuration planning includes network boundary definitions and any gateways in the networks.

To determine your network configuration, follow the steps listed below. For help with step 1, contact your HP representative. Steps 2 through 9 are described in this section.

1. Plan your network:

- Physical topology, or determining where the nodes and links in your network will be physically located. For example, if there are two nodes in a point-to-point network with heavy data communication traffic between them, you would probably install a direct link between those nodes. A direct link would enable these nodes to communicate more quickly without going through any intermediate nodes.

- Communication link types. Select your communication links according to the services that you want, geographical considerations, line speed requirements and any existing links. Communication links are described in the "Communication Links" subsection of the "Overview" section.

- Network boundaries. Determine how to group your nodes. Network topologies can be more efficient if nodes that often communicate with each other are grouped in the same network. For example, you may have a group of nodes that are used by an accounting department and a group of nodes used by a manufacturing department. Since the systems within each department primarily communicate with each other and seldom communicate with systems from the other departments, you could configure the systems so that there are two networks, with the network boundaries matching the division between the departments.

  If you have several networks, you may want to consider subnetting. Subnetting is an addressing technique for IP addresses in which you partition the host address into a subnet number and a node number. This allows several physical networks to share the same main network number and to form one logical network. All nodes in a network would share the same subnet number. And each network would have a different subnet number.

2. Construct network maps.

   A network map shows how nodes and networks are connected. When you draw this map, you do not need to include every node in the network, but you must include the following:

- the IP address for each network. Note whether each network is part of a subnetted network.
- all of the gateways in the network, if there are any.

   After you draw the overall network map, construct a map for each individual network. Include the following information in your network map:

- For each node, include the IP address, the node name, and the Router/1000 address, if there is one.
- For each gateway, show all of the IP addresses of the remote networks connected to that gateway.

3. Assign names to the nodes and determine other NS-ARPA nodal information—NS-ARPA file names and NS-ARPA services offered.
4. Assign LUs to the communication links.
5. Select Link Interface types.
6. Determine network boundaries.
7. Formalize network boundaries by assigning the following address information for the nodes:

- IP addresses
- Router/1000 addresses
- Guardian Nodes (only for networks with DS/1000-IV 91750 nodes)

8. Fill out routing information in a worksheet. A blank worksheet and examples are in Appendix A.
9. Assign Security Codes.

You should record the above information on the network map and on a Network Worksheet. You will be prompted for this information by NSINIT.

Network Map and Worksheet

A network map is a diagram of the links, nodes, and networks. A network map helps you configure your network and helps your HP representative should you have any problems with your network. The Network Worksheet contains information about the nodes and their configuration information.

Network Map

A network map should include the following information:

- node information—system type and resources (peripherals, amount of memory) and services supported
- link information—for LAN nodes, this can include the location of the coaxial cable, terminators, repeaters, AUI cables, and MAUs
- link interface types and network boundaries

In addition, you should mark the names and addresses that you assign the nodes. Network boundaries, names, and addresses are described in the following subsections. Figure 3-1 is an example of a network map.

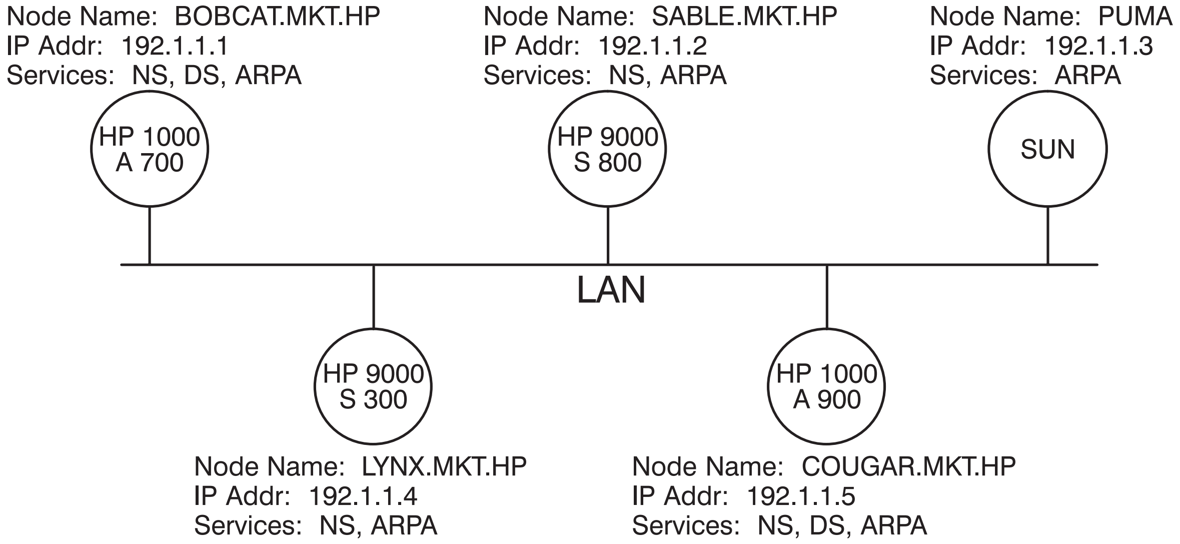


Figure 3-1. Sample Network Map

Network Worksheet

A Network Worksheet is used in conjunction with the Network Map to record information about the network and to help you generate and initialize the nodes in the network. The NS-ARPA/1000 product contains a blank Network Worksheet in the file /NS1000/EXAMPLES/NS_WORKSHEET.TXT, and a blank copy is printed in Appendix A of this manual.

The Network Worksheet provides space to record the following information:

- nodal information—the local node's name and address information, the names of the files used for configuration and initialization, services supported, and link and LU information
- routing information—information used in the NSINIT initialization dialogue to set up routing tables
- security information—security codes and ID sequences used to limit access to nodes.

Sample Network Worksheets are contained in /NS1000/EXAMPLES/EX_802_WORKSHEET.TXT and /NS1000/EXAMPLES/EX_RTR_WORKSHEET.TXT, and are listed in Appendix A of this manual.

Routing Worksheet

HP also provides a Routing Worksheet, which is a subset of the Network Worksheet. It duplicates the sections for local name and address information, and for routing information from the Network Worksheet. Blank Routing Worksheets are provided in Appendix A of this manual.

Sample Routing Worksheets are listed at the end of this section.

Node Names

For each node that supports NS Common Services and ARPA Services, you must assign a unique *node name*. The node name is used to uniquely identify nodes for NS Common Services and ARPA Services in a network. A fully-qualified node name consists of three fields, each separated by a period:

```
node [.domain][.organization]
```

When specifying node names, you *must* enter the delimiting periods. Each field can be up to 16 alphanumeric, underscore (_), or dash (–) characters; the first character must be alphabetic.

Domain and Organization Fields

After you have determined the network boundaries in your network, HP recommends that you use some convention when assigning domain and organization fields. For example, all nodes in the same network could have the same domain name, and all nodes in the same internet could have the same organization name.

For NS services (NFT and NetIPC) and ARPA services (TELNET and FTP), the domain and organization fields are optional. If a user does not supply domain and organization fields, the fields will default to the local node's fields.

Node names are often referred to as *host names* in other ARPA systems. Currently NS-ARPA/1000 does not support *ARPA domain names*.

Transparent File Access

If you support Transparent File Access (also called DS File Transparency), HP recommends that you assign node names in the following manner:

- Assign names so that the node fields are unique throughout the internet.
- Use the node field of the NS-ARPA node name as the system name in the file `NODENAMES::SYSTEM` (the file used by Transparent File Access).

For more information about installing Transparent File Access software, refer to *RTE-A System Generation and Installation Manual*, part number 92077-90034.

Worksheet Entry

Enter the node name for each node in Section 1 of the Network Worksheet (or the Routing Worksheet).

NS-ARPA Files

The Files section of the Network Worksheet (Section 4) lets you record the files used by the NS-ARPA network utilities. For NSINIT and NRINIT, enter the input files from which those utilities are to read configuration information. For FMTRC (the trace formatting utility), enter the input file that is to contain the raw trace information that FMTRC will format. For NSTRC (the trace utility) and EVMON (the event logging utility), enter the files to which the utilities are to send output. All default files are listed in parentheses on the worksheet.

NS-ARPA Services

The Services section of the Network Worksheet (Section 7) lets you record the optional NS-ARPA services supported at each node. Note that all NS-ARPA/1000 nodes support NetIPC. The other parts of Section 7 of the Network Worksheet are as follows:

- `NFT` is Network File Transfer.
- `DS/1000-IV Compatible Services` are divided into HP 1000—HP 1000 services (DEXEC, PTOP, REMAT, RDBA, RFA, TRFAS, Utility subroutines, Remote I/O Mapping, Remote System Download, and Remote Virtual Control Panel) and HP 1000—HP 3000 services (PTOP, RFA, RMOTE, and Utility subroutines).

The worksheet also lists the monitors required for DS/1000-IV Compatible Services. For descriptions of the DS/1000-IV Compatible Services and monitors, refer to the “Internal Resources” section, subsection “Software Modules.”

FTP, TELNET, and RPM are automatically configured for you and do not appear on the worksheet.

Link Interfaces

Link Interfaces (LIs) act as interfaces between the Layer 3 protocol and the communication link drivers. The LI software determines which communication link type to use and then passes outbound messages to the appropriate driver for the specific communication link.

Communication links are bound to link interface types at initialization time. Several types of communication links can be bound to the same LI type. In NS-ARPA/1000, there are three LI types:

- *ROUTER/1000 (RTR)*
- *IEEE 802.3 LAN (802)*
- *Ethernet LAN (ETHERNET)*

If you have both IEEE 802.3 and Ethernet LAN, then the LI type is LAN.

Table 3-1 lists the communication links and corresponding link interface types.

Table 3-1. Communication Links and Link Interface Types

Communication Link	System Connection from NS-ARPA/1000 to:	Link Interface Types
IEEE 802.3 LAN	NS-ARPA/1000 NS3000/V NS3000/XL NS/9000 NS for the DEC VAX computer OfficeShare on the PC ARPA/1000 ARPA/9000	IEEE 802.3 (802)
Ethernet LAN	NS-ARPA/1000 NS/9000 ARPA/1000 ARPA/9000 ARPA Services/Vectra ARPA on SUN workstations	Ethernet (ETHERNET)
HDLC	NS-ARPA/1000	Router/1000 (RTR)
	DS/1000-IV	Router/1000 (RTR)
BISYNC	NS3000/V DS/3000	No link interface type
X.25	NS-ARPA/1000	Router/1000 (RTR)
	DS/1000-IV	Router/1000 (RTR)
	NS3000/V DS/3000	No link interface type

Network Boundaries

Network types and network boundaries are largely defined by LI types. The LI type used to connect the nodes in a network defines the network type. The network type is also related to the network protocol used to route messages within the network. NS-ARPA/1000 supports two network types: Router/1000 and LAN (IEEE 802.3 and Ethernet).

Note Network types and boundaries are also affected by node types. Although NS-ARPA nodes can communicate with DS/1000-IV (91750) nodes and HP 3000s that support only DS/1000-IV Compatible Services, *these nodes do not support NS Common Services and ARPA Services*. The configuration requirements for these nodes differ from the requirements for NS-ARPA nodes. The configuration information about these nodes that you must supply is described in a later subsection.

The following subsections describe how LI types define the gateways and boundaries between the networks. When determining network boundaries, *disregard Bisync or X.25 links to HP 3000 nodes*.

IEEE 802.3 and Ethernet LIs

Within the same LAN, nodes configured to handle only Ethernet packets cannot communicate with nodes configured to handle only IEEE packets. If a node supports both Ethernet and IEEE 802.3, then that node can communicate with the other two types of nodes.

For IEEE 802.3 and Ethernet LIs, network boundaries are determined by the following rules:

- *All nodes in a single IEEE 802.3 or Ethernet network must be connected to the same bus*. Therefore, an *IEEE 802.3 or Ethernet network* is defined as all the nodes connected to the same bus. If a node has LAN links to two buses, it belongs to two networks and is a gateway. There is a one-to-one mapping between the number of LAN interface cards in a node and the number of LAN networks to which the node belongs.
- *All links that connect nodes in a single network must have the same LI type*. A node with a 802 and a RTR LI belongs to two networks and is a gateway. This is true even if the RTR LI is connected to a DS/1000-IV node. In Figure 3-2, Node 5 belongs to Network A and to Network B and is a gateway.

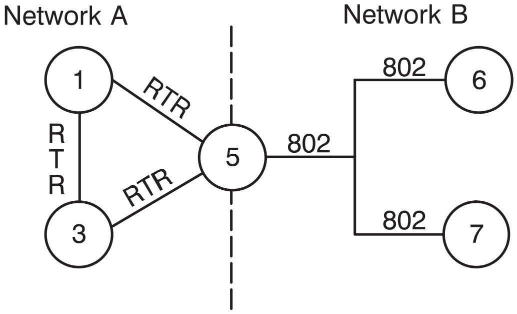


Figure 3-2. Internet Showing a Gateway

Router/1000 LIs

For RTR LIs, network boundaries are determined by the following rules:

- *All links that connect nodes in a Router/1000 network must have a RTR LI type*. A *Router/1000 network* is defined as a group of nodes that are connected by RTR LIs. (There may be redundant, non-RTR LIs in a Router/1000 network, but for any two nodes in the network, a route must exist between them that consists only of RTR LIs.) The Router/1000 network can include DS/1000-IV nodes.
- Therefore, *all nodes connected by RTR LIs belong to the same network*. This is true even if some of the nodes are DS/1000-IV (91750) nodes.
- *All links that connect nodes in a single network must have the same LI type*. A node with a RTR and a 802 LI belongs to two networks and is a gateway.
- *A node with only RTR LIs belongs to only one network, no matter how many RTR LIs it has*.

Gateways

A node is a gateway if it meets one or both of the following conditions:

- the node has a RTR and an 802, ETHERNET, or LAN LI
- the node has links to more than one LAN bus.

LAN Gateways

LAN gateways (nodes with links to more than one LAN bus) may become congested during periods of heavy internetwork traffic between the the LANs. In such situations, the LAN link speed may cause the NS-ARPA inbound and outbound message handlers, INPRO and OUTPRO, to be continuously executing, using all available CPU cycles. This prevents all programs with lower priorities from executing.

Networks

As stated above, NS-ARPA/1000 supports two network types: LAN (802, Ethernet, or both) and Router/1000.

A LAN network is defined as all the nodes connected to the same LAN bus.

A Router/1000 network is defined as a group of nodes that are connected by RTR LIs. (There may be redundant, non-RTR LIs in a Router/1000 network, but for any two nodes in the network, a route must exist between them that consists only of RTR LIs.)

Redundant Links

You can configure redundant links between NS-ARPA/1000 nodes so that there are multiple routes between nodes. However, NS-ARPA/1000 will use alternate routes only in Router/1000 networks with dynamic rerouting. In all other cases, NS-ARPA/1000 always attempts to use the same route between nodes, even if that route is currently unusable.

NS-ARPA/1000 software routes messages for DS/1000-IV Compatible Services (RTE-RTE) through RTR LIs whenever possible. In topologies with redundant, non-Router/1000 links between nodes, NS-ARPA/1000 always routes messages for DS/1000-IV Compatible Services (RTE-RTE) over the RTR LIs, even if there are direct, non-RTR links between the source and destination nodes. This is true even if the intermediate nodes are DS/1000-IV nodes. For example, in Figure 3-3, messages for a TRFAS transaction (or any other RTE-RTE DS/1000-IV Compatible Service) between nodes A and B would be routed through nodes C and D, rather than through the LAN network.

Messages for NS Common Services (such as NFT) between nodes A and B would still be routed through the LAN network.

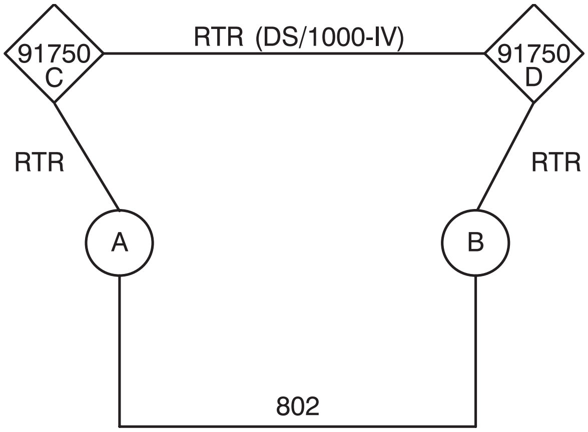
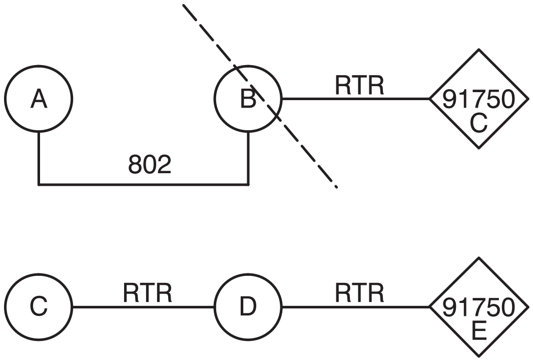


Figure 3-3. Networks with Redundant Links

DS/1000-IV (91750) Nodes

Links to 91750 nodes must have RTR LI types. If a node is a gateway between a LAN network and a RTR network, it will have two LI types (802, Ethernet, or LAN; and RTR). For example, in Figure 3-4, Node B is a gateway between an 802 network and a RTR network.



The network boundary is indicated by a dotted line.

Figure 3-4. Two Networks with 91750 Nodes

DS/1000-IV Compatible Services (RTE-MPE) Links

Bisync and X.25 links used for DS/1000-IV Compatible Services (RTE-MPE) do not have a link type. You cannot construct a network with these links because they do not support store-and-forward, Dynamic Rerouting, or IP routing. In Figure 3-5, Nodes A and B are HP 1000 nodes that have Bisync links to an HP 3000, Node C; Nodes A and B cannot communicate with each other through Node C. HP recommends that you use such links for leaf nodes, that is, nodes not used for store-and-forward traffic.

Do not configure IP information for HP 3000s that do not support NS Common Services or ARPA Services.

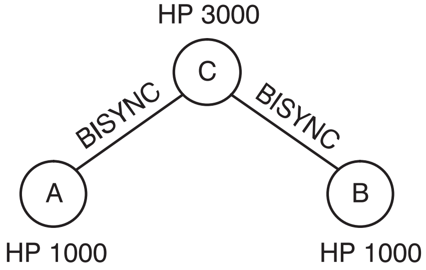


Figure 3-5. Nodes A and B Cannot Communicate

Worksheet Entry

Enter the LI types in Section 5 of the Network Worksheet, and the number of networks that each node belongs to in Section 6 of the Network Worksheet.

Link LUs

For each communication link, assign two consecutive LUs. The first (lower) LU is the transmit LU. The second (upper) LU is the receive LU. For LAN/1000 links, the transmit LU (also called the user LU) must be an even number.

For more information on specifying LU assignments, refer to the section "System Generation."

Worksheet Entry

Enter the transmit LUs in Section 5 of the Network Worksheet. For X.25 links used for DS/1000-IV Compatible Services (RTE-MPE), enter the pool LUs at the end of Section 5.

Addresses

After marking the nodes and network boundaries on the network map, you can assign addresses to the nodes.

You must configure the following information:

- *IP Addresses*, which are used by NS Common Services, ARPA Services, and IP.
- *Router/1000 Addresses*, which are used by DS/1000-IV Compatible Services and Router/1000 links (HDLC, Multidrop, and X.25 links to HP 1000s).
- *Guardian Nodes* (only for networks with DS/1000-IV 91750 nodes).
- *Nodal Registry Configuration*. Nodal Registry configuration is *not* necessary when all the nodes in a network are all IEEE 802.3 LANs and are all HP systems.

The following subsections describe how to configure and record the above information on the network map and on the Network Worksheet (Appendix A).

IP Addresses

IP addresses are used by NS Common Services, ARPA Services, and IP. You do not assign IP addresses to 91750 nodes or HP 3000s that do not support NS Common Services or ARPA Services.

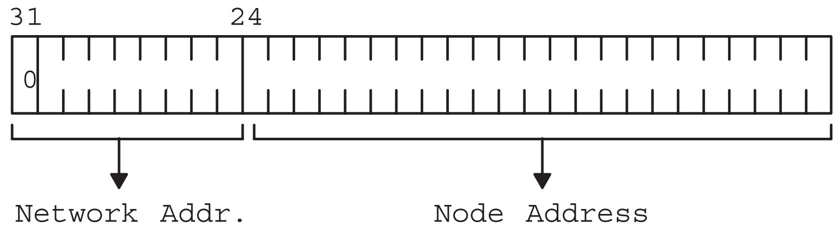
An IP address consists of two parts: a *network address*, which identifies the network; and a *node address*, which identifies a node within a network. A network address is concatenated with a node address to form the IP address and uniquely identify a node within a network within an internet.

If you have several networks, you may consider using subnetting. In this case, an IP address has three parts: a network number, a *subnet* number, and a node number. Using subnet addressing is optional. Refer to “IP Addresses with Subnetting” after reading the following IP addressing information.

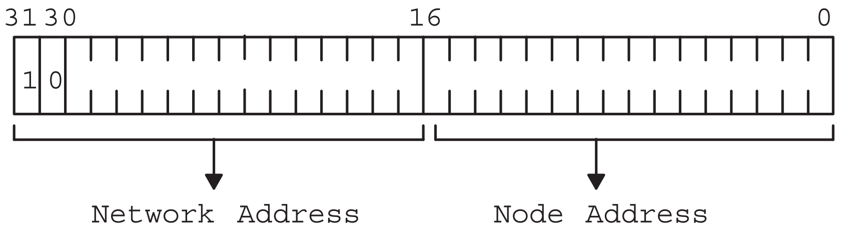
IP Address Formats

There are three IP address classes, each accommodating a different number of network and node addresses. The address classes are defined by the most significant bits of the address, as follows:

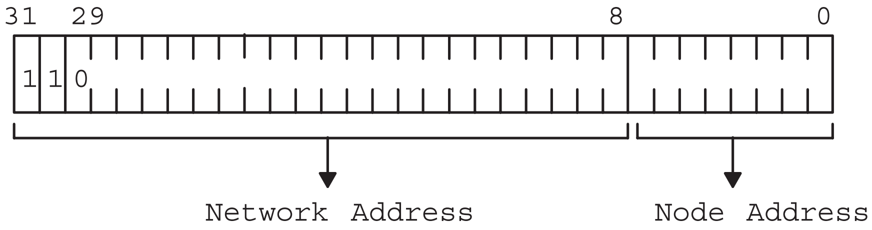
Class A address—



Class B address—



Class C address—



The address classes can also be broken down by address ranges. IP addresses are typically represented by converting the bits to decimal values an octet (8 bits) at a time, and separating each octet’s decimal value by a period (.). Therefore, IP addresses are typically of the following form:

nnn.nnn.nnn.nnn

where *nnn* is a number from 000 to 255, inclusive.

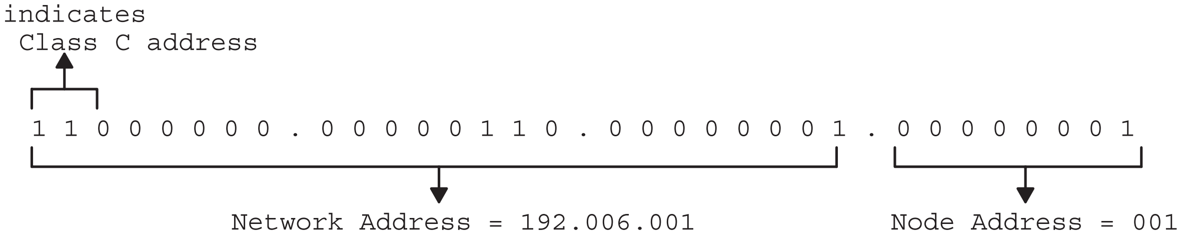
Table 3-2 lists the number of networks and nodes and the address ranges for each address class.

Table 3-2. IP Address Classes

Class	Networks	Nodes per Network	Address Range
A	127	16777215	000.000.000.000*—127.255.255.255
B	16383	65535	128.000.000.000—191.255.255.255
C	2097151	255	192.000.000.000—223.255.255.255
Reserved	–	–	224.000.000.000—255.255.255.255*

* Do not assign the network and node addresses to all zeros or all ones; they are reserved. The address of all ones (255.255.255.255) is used internally for broadcasting.

To determine a network address and node address from an IP address, you must separate the network and node address fields. For example, the bit representation of IP address `192.006.001.001` is separated as follows:



IP Addresses with Subnetting

Subnetting is an optional addressing scheme that allows you to partition the *node address* portion of an internet address (IP address) into discrete *subnetworks*. The term, *subnetted network*, is sometimes used for a subnetwork.

The node address is divided into a *subnet number* and a *node number* to identify the node within this subnetwork. The *network address* portion of the IP address remains unchanged.

```
IP Address = Network Address + Node Address
           = Network Address + (Subnet Number + Node Number)
```

A subnetted network can communicate with a non-subnetted network.

Why Use Subnetting

Subnetting allows you to use one network address for two or more physically distinct networks. For example, suppose a large installation has many interconnected nodes all on the same physical network. Hardware configuration restrictions or performance degradation might occur with so many nodes on the same physical network. With subnetting, you can install several smaller physical networks (connected via gateways) and have them all share the same network address. You then use different subnet addresses for each of these physical networks. Each network would actually be a *subnetwork*. In summary, a network address would identify a group of networks, and the subnet numbers identify the subnetworks.

Because subnetting allows you to use fewer network addresses, you may hide the internal structure of your company's networks. So, instead of assigning different network addresses to each physical network, a company needs only one network address. People outside the company need only know one network address to be able to send to any node in the company.

How to Use Subnetting

Subnet numbers are used like network numbers to distinguish each subnetwork. All nodes on the same physical LAN are assigned the same subnet number. Nodes on a different LAN have a separate subnet number. If the two networks are connected via a gateway and are to be subnetted, then each node in both networks will share the same network address.

Determining the number of subnetworks and the number of nodes in each subnetwork depends on how many bits are used in the node address portion of the IP address. For example, a Class C IP address uses 8 bits for the *node address* portion. Of these 8 bits, if 3 bits are used for subnet numbers, then we can have 7 subnetworks and 29 nodes per subnetwork. Node addresses of all zeros (000.000.000.000) or all ones (255.255.255.255) are reserved and are not allowed.

Class C address—

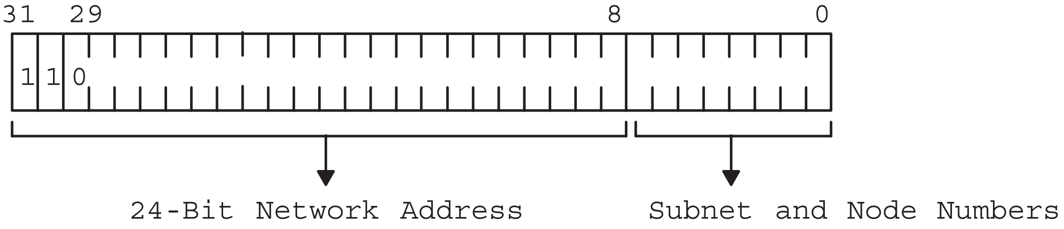


Table 3-3 lists the valid IP addresses for up to 7 subnetworks and 29 nodes per subnetwork.

Table 3-3. Subnet Addressing Example

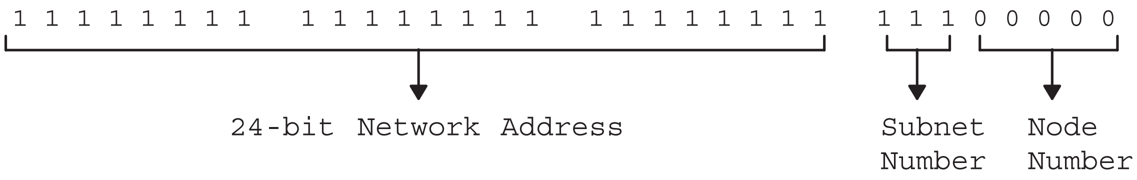
Subnet Address (binary)	Subnet Address (decimal)	IP Address Range
000	0	000.000.000.000 is reserved and cannot be used
001	1	n.n.n.33—n.n.n.62
010	2	n.n.n.65—n.n.n.94
011	3	n.n.n.97—n.n.n.126
100	4	n.n.n.129—n.n.n.158
101	5	n.n.n.161—n.n.n.190
110	6	n.n.n.193—n.n.n.222
111	7	n.n.n.225—n.n.n.254
		255.255.255.255 is reserved and cannot be used; it is used internally for broadcasting

Subnet Masks

The vital part of subnetting is the 32-bit subnet mask. The subnet mask specifies the portion of the node address used to identify the subnet number. The remaining part of the node address is used to identify the node on that particular subnetwork.

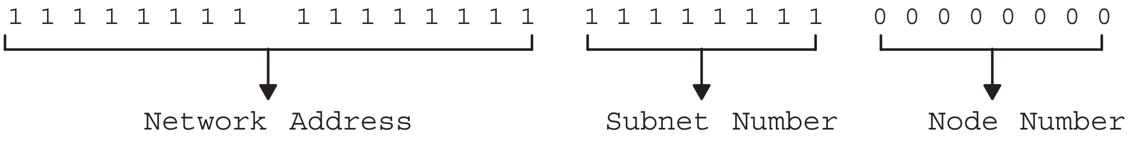
Bits, in the subnet mask, are set to all zeroes for the node number and set to all ones for the network address and subnet number. For the above example of a Class C IP address, the subnet mask would be as follows:

Class C address with subnet mask of 255.255.255.224 decimal—



In another example, given a Class B IP address with the following subnet mask means that the first two octets of the IP address identifies the main network, the third octet is the subnet number and the fourth octet identifies the node number:

Class B address—



All nodes in a subnetwork must use the same subnet mask. A subnetted network will communicate with a non-subnetted network.

Subnetted Network Example

The following example shows four subnetworks sharing a Class C address of 192.006.012.

Subnetwork A nodes: 192.006.012.034 and 192.006.012.035

gateway node: G1 at 192.006.012.033 and 192.006.012.129
Subnet number = 1 (001 in binary)
Node address range = 33-62
Subnet mask = 255.255.255.224

Subnetwork B nodes: 192.006.012.067 and 192.006.012.066

gateway node: G2 at 192.006.012.065 and 192.006.012.130
Subnet number = 2 (010 in binary)
Node address range = 65-94
Subnet mask = 255.255.255.224

Subnetwork C nodes: 192.006.012.098 and 192.006.012.099

gateway node: G3 at 192.006.012.097 and 192.006.012.131
Subnet number = 3 (011 in binary)
Node address range = 97-126
Subnet mask = 255.255.255.224

Subnetwork D gateway nodes: G1, G2, and G3 at 192.006.012.129, 192.006.012.131, and 192.006.012.130

Subnet number = 4 (100 in binary)
Node address range = 129-158
Subnet mask = 255.255.255.224

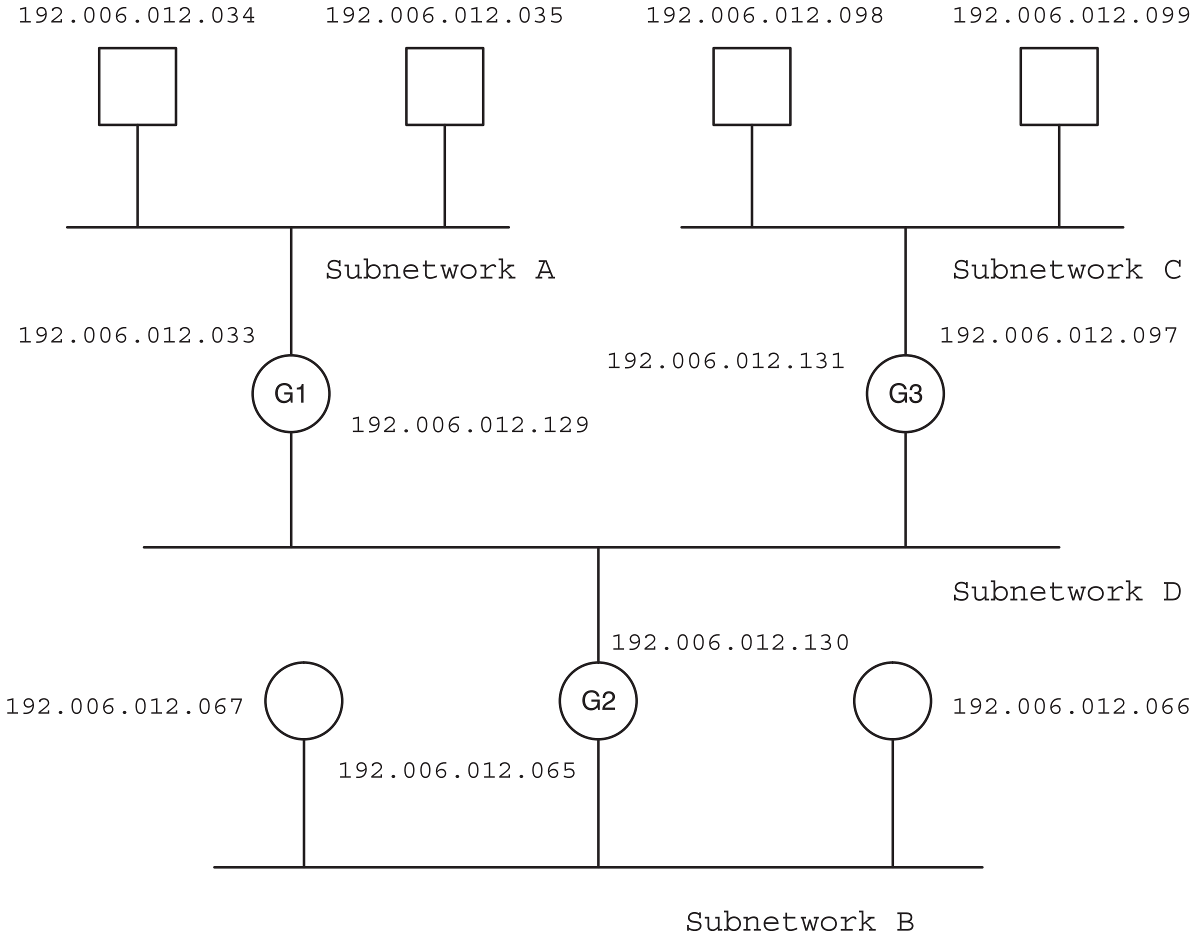


Figure 3-6. Subnetted Network Example

Assigning IP Addresses

You must assign an IP address as follows:

- You must assign an IP address for each NS-ARPA node in your network *except* nodes that do not support NS Common Services or ARPA Services.
- Do not assign any nodes with a reserved address of all zeros or all ones (see Table 3-2).
- If a node is a member of more than one network, you must assign that node an IP address for each network of which it is a member.

When assigning IP addresses, you must determine network and node addresses, as described below.

Network Addresses

To assign IP network addresses, follow these rules:

- Using the rules described in the Network Boundaries subsection, determine how many networks you have and where your network boundaries are. (The number of networks that each node belongs to should be entered in Section 6 of the Network Worksheet.)
- Each network has a unique network address throughout the internet.
- All nodes in a network must have the same network address.
- If a node belongs to more than one network, it must have one and only one IP address for each network to which it belongs. Note that since a node can belong to only one Router/1000 network, a given node could have at most one IP address associated with a Router/1000 network.
- Do not assign any networks with a reserved network address of all zeros or all ones (see Table 3-2).

Obtaining Network Addresses

HP has obtained a block of Class C network addresses from DARPA to assign to HP customers. You can obtain Class C addresses that are unique within the ARPANET by contacting HP at the following address:

Network Administration Office, Dept. NET
Information Networks Division
Hewlett-Packard Company
19420 Homestead Road
Cupertino, California 95014

Note Although any address assigned from HP is unique within DARPA's ARPANET, this does not imply that your system is compatible with nor supported on the ARPANET.

If you attempt to communicate with external networks, you should consider the security risks of external networks accessing your network.

The IP addresses used in this manual are given as examples only. Do not use these addresses in your network.

Node Addresses

To assign IP node addresses, follow these rules:

- Node addresses must be unique within each network, but they do not have to be unique throughout the internet. For example, you could have a node with node address 55 in Network 18 and a node with node address 55 in Network 21. You can assign node addresses according to your own needs, but they must be within the ranges for the IP address class that you are using.
- Do not assign any nodes with a reserved node address of all zeros or all ones (see Table 3-2).
- If nodes support DS/1000-IV Compatible Services (RTE-RTE) or have RTR LIs, you may want to assign node addresses that are unique throughout the network and that correspond to the Router/1000 addresses. Router/1000 addresses are described later in this section.

DS/1000-IV (91750) Nodes

Although you do not assign IP addresses to 91750 nodes, 91750 nodes may affect network boundaries and IP addresses as follows:

- As noted in the "Network Boundaries" subsection, all links connected to 91750 nodes must be RTR LIs. Therefore, if a node has a 802 LI and a link to a 91750 node, that node is a gateway and belongs to two networks. Therefore, you must assign two IP addresses to that node, one for each LI. The IP addresses must have different network addresses. For example, in Figure 3-7, Node B belongs to two networks, and requires two IP addresses with different IP network addresses.

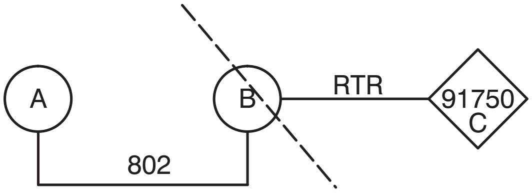


Figure 3-7. Node B is a Gateway

- All NS-ARPA/1000 nodes connected to the same network of 91750 nodes must have the same IP network address. For example, in Figure 3-8, Node B is a gateway; the network portion of the IP address associated with Node B's Router/1000 link and the network portion of the IP addresses for Node C, Node D, and Node E must match. Node E is a gateway to a third network.

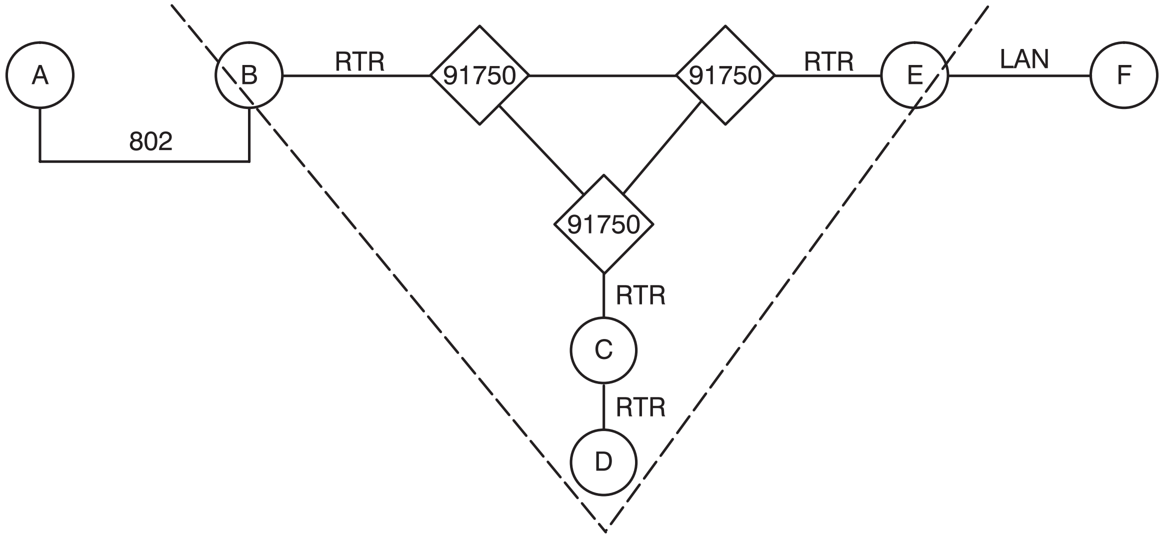


Figure 3-8. Three Networks, Including a Network with 91750 Nodes

Worksheet Entry

Enter the IP addresses in Section 2 of the Network Worksheet (or in the Routing Worksheet).

Router/1000 Addresses

The Router/1000 address is used by DS/1000-IV Compatible Transport and the Router/1000 software; it is unrelated to the IP address.

You must assign Router/1000 addresses to the following types of nodes:

- nodes with ROUTER LIs
- nodes that offer or use DS/1000-IV Compatible Services, including 91750 nodes and nodes without RTR LIs.

A Router/1000 address must be an integer between 1 and 32767, inclusive.

Note that there may be more than one Router/1000 network in an internet. Router/1000 addresses must be unique within the internet (not just within the individual networks). This is necessary because DS/1000-IV Compatible Services use the Router/1000 address to uniquely identify nodes throughout the internet.

You may want to assign Router/1000 addresses so that they match the node portion of the IP address for a given node.

On DS/1000-IV (91750) nodes, the Router/1000 address is referred to as the *node number*.

Worksheet Entry

Enter the Router/1000 addresses in Section 3 of the Network Worksheet (or in the Routing Worksheet).

Guardian Nodes for DS/1000-IV (91750) Nodes

If you have DS/1000-IV (91750) nodes and an internet (you have more than one network), you must assign NS-ARPA/1000 *guardian* (91790) nodes.

NS-ARPA/1000 nodes that are guardian nodes allow DS/1000-IV (91750) to communicate with nodes on remote networks. The guardian node is a gateway into the DS/1000-IV network. Guardian nodes remove IP headers from messages sent over non-Router/1000 links before delivering them to the DS/1000-IV (91750) node. Guardian nodes also add IP headers to messages sent from DS/1000-IV (91750) nodes that are to be forwarded to remote networks.

You specify the guardian node by entering the IP address of the guardian node in the Nodal Routing Vector (NRV) entry for the DS/1000-IV node. Thus, there are three nodes to consider when assigning guardian nodes:

- the NS-ARPA/1000 node you are configuring (hereafter referred to as the *local node*)
- the *DS/1000-IV node*
- the NS-ARPA/1000 *guardian node*.

Assigning Guardian Nodes

Assign an NS-ARPA/1000 guardian node if

- the local NS-ARPA/1000 node and the DS/1000-IV (91750) node are not members of the same directly connected network.

  In other words, you must assign a guardian node if the local node uses any non-Router/1000 links to reach the DS/1000-IV node being entered into this NRV.

Do *not* assign an NS-ARPA/1000 guardian node if

- the local NS-ARPA/1000 node and the DS/1000-IV (91750) node are members of the same DCN.

Specifying Guardian Nodes

You select and specify the guardian node in the NRV as follows:

- The guardian node will be a gateway into the Router/1000 network that contains the DS/1000-IV node.

  This is because the guardian node and the DS/1000-IV (91750) node must be members of the same Router/1000 network. That is, all links between the DS/1000-IV (91750) node and its guardian must have RTR LIs.

For example, in Figure 3-9, at Node A you would specify Node B's IP address in the NRV entry for Node D. All messages between Nodes A and D would pass through Node B, and Node B would remove or add the IP headers to the messages. Note that at Node C you would not assign a guardian node for Node D, since Node C and Node D belong to the same Router/1000 network.

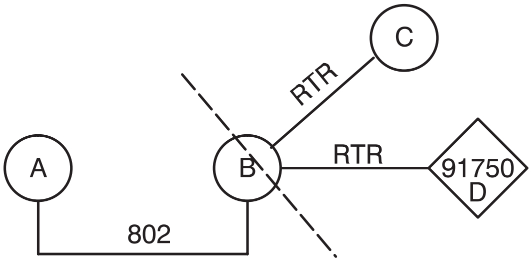


Figure 3-9. Node A Would Assign Node B as Node D's Guardian

Worksheet Entry

Specify the guardian node in the Router/1000 NRV (Section 11 of the Network Worksheet), or in the Routing Worksheet.

LAN Station Addresses

Hewlett-Packard assigns a *station address* to each LAN/1000 interface card during manufacturing. The station address is used for addressing within the LAN, and has the following format:

```
hh-hh-hh-hh-hh-hh
```

where `h` is a hexadecimal digit (0-F).

The first six digits of all LAN cards manufactured by HP are `08-00-09`, so all HP LAN cards have addresses of the following format:

```
08-00-09-hh-hh-hh
```

To determine the HP-assigned station address of your card or cards, refer to *HP 12076A LAN/1000 LANIC Link Installation Manual*.

Assigning Station Addresses

HP recommends that you use the HP-assigned station address of the LAN interface card. However, if you want to assign your own addresses, read this subsection and the HP 12076A LAN/1000 Link Node Manager's Manual.

To assign your own LAN/1000 station addresses, you must conform to the IEEE standard. NS-ARPA/1000 supports IEEE 802.3 48-bit (six-octet) station addresses, which have the following format:

msb							lsb (transmitted first)	
7	6	5	4	3	2	1	0	
MA	MA	MA	MA	MA	MA	U/L	0	Most significant Byte (transmitted first)
MA	MA	MA	MA	MA	MA	MA	MA	
MA	MA	MA	MA	MA	MA	MA	MA	
CA	CA	CA	CA	CA	CA	CA	CA	
CA	CA	CA	CA	CA	CA	CA	CA	
CA	CA	CA	CA	CA	CA	CA	CA	

The bits in the station address have the following meanings:

CA = Card Address bits comprising the lower six hex digits
MA = Manufacturer's Address bits. Bit 1 of the most significant octet has the following meaning:

```
U/L = 0  Globally Administered Address
    = 1  Locally Administered Address
```

The U/L (Universal Product Code/Local) bit determines whether the manufacturer's address is globally administered, or locally administered. A *globally administered address* implies a universally unique address as administered by the IEEE. When shipped from the factory, each Hewlett-Packard LANIC card contains such an address: a unique Manufacturer's Address (08 00 09 hex) assigned to Hewlett-Packard, and the lower six hex digits assigned by Hewlett-Packard. (Contact the IEEE for a unique Manufacturer's Address.)

A *locally administered address* is controlled by the user, and is not guaranteed to be universally unique.

Bit 0 of the most significant octet is the I/G bit, which must be set to 0. This defines the address as an individual address, rather than a group address. An *individual* address is associated with a particular station on the network and implies that a single LANIC card is addressed, rather than a broadcast or multicast address. A broadcast address is used to address all stations, or nodes, on a LAN and a multicast address is used to address a specific group of nodes.

The IEEE standard states that addresses must be transmitted in octets, least significant bit first. Because of the restrictions on bit 0 of the first octet, the second hexadecimal digit of station addresses *must* be 0, 2, 4, 6, 8, A, C or E. The following addresses are all valid station addresses:

```
12-68-9D-1C-7F-1A
34-99-BE-D1-C7-FF
36-11-FA-B5-2E-88
08-00-09-11-22-33
5A-99-BE-D1-C7-FF
AC-99-BE-D1-C7-FF
7E-59-53-42-28-08
```

For more information on IEEE 802.3 addresses, refer to *HP 12076A LAN/1000 Link Installation Manual* or *HP 12076A LAN/1000 Node Manager's Manual*.

Worksheet Entry

Enter the station address for each LAN card in Section 5 of the Network Worksheet.

Nodal Registry Configuration

This subsection describes the items you must configure for Nodal Registry.

Nodal Registry Features

The Nodal Registry is software that manages information that the transport and services use to establish connections with remote nodes. The Nodal Registry stores this information in DSAM as *Nodal Path Reports* (NPRs). Nodal Path Reports map node names to address information. Nodal Path Reports are indexed according to node name, and each contain an IP address (or addresses, if the node belongs to multiple networks). If the node has IEEE 802.3/Ethernet links, the NPR can also contain a station address for each such link. In addition, an NPR denotes which protocols and NS Common Services and/or ARPA Services the node supports.

You need to initialize the Nodal Registry to communicate with nodes that cannot reply to Probe requests, such as

- non-HP nodes
- nodes not on the same IEEE 802.3 LAN. By having a Probe proxy server on the LAN, even this situation can avoid initializing the Nodal Registry.

If you need to initialize the Nodal Registry, run NRINIT, which is described in Section 11. The local node is automatically put into the Nodal Registry by NSINIT. Refer to the “Network Management” subsection in the “NSINIT” section of this manual. If you try to configure an NPR for the local node, NSINIT issues a warning message and ignores that NPR.

Probe Features

The Probe protocol allows IEEE 802.3 nodes to query one another for NPRs and other address information. The Probe protocol provides the following features among HP computers only:

- node name to Nodal Path Report mapping
- IP address to IEEE 802.3 station address mapping
- nodal Registry Updates.

Configuration

You configure the following information for Nodal Registry and Probe:

- Probe Proxy Servers
- Nodal Path Reports

In addition, you can configure Probe Addresses. *However, HP recommends that you use the default Probe Addresses*.

Probe Proxy Servers

Probe Proxy Servers can provide NPRs for any node in the internet. If an IEEE 802.3 LAN has a Probe Proxy Server, all nodes on that IEEE 802.3 LAN can get the NPRs they need from the Probe Proxy Server. At the other nodes on the LAN, you only need to configure an NPR for that node which is automatically done by NSINIT.

HP recommends that you configure at least one Probe Proxy Server on each LAN, as follows:

> At the Probe Proxy Server node, configure NPRs for all the nodes in the *internet, including non-LAN nodes*, all Ethernet LAN nodes, and all non-HP nodes.
>
> Do not include NPRs for the other nodes on the IEEE 802.3 LAN, since all nodes on the same IEEE 802.3 LAN can get NPRs for each other via Probe Name Requests.

If the Probe Proxy Server is also a gateway between LANs, it can serve as a Probe Proxy Server for all the LANs to which it belongs.

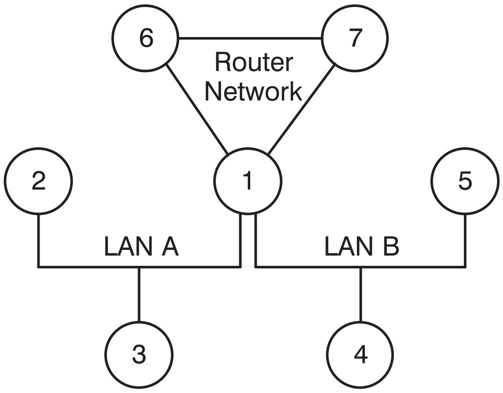


Figure 3-10. Node 1 Could Be the Probe Proxy Server for LAN A and LAN B

For example, in Figure 3-10, it would be most convenient to designate Node 1 as the Probe Proxy server. Node 1 can serve two LANs since it is the gateway between them.

Nodal Path Report Configuration

You configure the maximum number of NPRs for a given node via NSINIT; you configure NPRs via NRINIT. For the NPR syntax, refer to the section “Nodal Registry Configuration” of this manual.

The local node is automatically put into the Nodal Registry by NSINIT. Refer to the “Network Management” subsection in the “NSINIT” section of this manual. If you try to configure an NPR for the local node, NSINIT issues a warning message and ignores that NPR.

You configure Nodal Path Reports as follows:

- If the local node does not have an 802 LI, you must configure Nodal Path Reports for all the nodes in the *internet*, except any nodes that do not support NS Common Services or ARPA Services.

- If the local node is a Probe Proxy Server or is a member of a LAN without a Probe Proxy Server, you must configure NPRs for all the nodes in the *internet*, except any nodes that you do not want to communicate with or do not support NS Common Services or ARPA Services.
- If the local node is a member of a LAN with a Probe Proxy Server and is not a Probe Proxy Server, configuring NPRs for remote nodes is optional; the local node can use Probe to query the Probe Proxy Server for NPRs. However, using Probe to get an NPR takes more time than looking up a configured NPR.

For example, at Node 1 in Figure 3-10, you would configure NPRs for every node in the internet except for Node 1. At nodes 2, 3, 4 and 5, no NPRs are required to be configured. The local nodes are automatically configured by NSINIT. At nodes 6 and 7, you must configure NPRs for all the remote nodes in the internet, since nodes 6 and 7 are Router/1000 nodes and cannot use Probe.

Probe Addresses

With IEEE 802.3 networks, messages are transmitted to every node on the IEEE 802.3 LAN. To determine which messages to receive, each LAN node maintains a list of IEEE 802.3 addresses, called an Inbound Address List. A given node will receive only messages with a destination address that matches an address on its Inbound Address List. Each node's Inbound Address List contains the node's station address. In addition, a group of nodes may have the same *multicast addresses* in their Inbound Address List. Messages with that multicast address would be received by all nodes in that group. Probe uses multicast addresses to send messages to HP nodes in a LAN. At each LAN node, you configure two Probe multicast addresses: a Target Address and a Proxy Address. For NS-ARPA, the default Target Address is:

09-00-09-00-00-01

The default Proxy Address is:

09-00-09-00-00-02

Note If you have LAN/1000 Node Manager software and the appropriate multicast address file for your card (the file `/files802/addr`*hhhhhhhhhhhh*`/mcast.txt`, where *hhhhhhhhhhhh* is the card's station address) exists on your system with data in it, *the multicast addresses for NS-ARPA must be included in the multicast address file*. Note that you do not need a multicast address file for your card if the NS-ARPA/1000 multicast addresses are the only multicast addresses for your system.

HP recommends that you use the default multicast addresses, unless a non-HP node requires that you use different multicast addresses. If you do not use the default addresses, you must configure the same Target Addresses and Proxy Addresses on all nodes in a given LAN.

The format of the address must meet the specifications for multicast addresses, as described in *HP 12076A LAN/1000 Link Installation Manual* and *HP 12076A LAN/1000 Node Manager's Manual*.

Target Addresses

Probe uses the configured Target address for two addresses: the Inbound Target Address and the Outbound Target Address. The Inbound Target Address is added to every LAN node's Inbound Address List at initialization time, and allows the LAN nodes to receive Probe Name Requests. LAN nodes use the Outbound Target Address to send Probe name and address requests (Name Requests and Virtual Address Requests).

Proxy Address

Probe uses the configured Proxy address for two addresses: an Inbound Proxy Address and an Outbound Proxy Address. The Inbound Proxy Address is used to receive Proxy Name Requests, and is added to the Inbound Address List of Probe Proxy Servers only. All nodes on the LAN use the Outbound Proxy Address to send Proxy Name Requests.

Worksheet Entry

In Section 8 of the Network Worksheet, indicate whether or not the node will be a Probe Proxy Server. If a node is to be a Probe Proxy Server, enter the Target and Proxy addresses (the defaults are listed on the worksheet).

HP 3000 Computers with DS/1000-IV Compatible Services

For HP 3000 computers without NS Common Services, do not assign any IP or Router/1000 addresses. However, if you have a dial-up Bisync link to such a node, you may want to assign ID sequences to limit access to the nodes. ID sequences are described later in this section, in the "Security Codes" subsection.

Routing Information

You must configure routing information at each node in the network via NSINIT. You can enter this information in the Routing section of the Network Worksheet or in the Routing Worksheet. You may need to use several worksheets for your internet.

Note When discussing routing information, this manual uses the term *local node* to indicate the node for which you are configuring routing information. The local node need not be the node at which you are physically located.

You must configure the following routing information at each node:

- IP Directly Connected Network (DCN)
- IP Gateway Table (GT)
- Router/1000 Nodal Routing Vector (NRV)

IP Routing Information

IP uses the DCN and GT information to route messages throughout the internet as follows:

- The IP DCN (Directly Connected Network) indicates all the Router/1000 and IEEE 802.3/Ethernet/LAN networks to which the local node belongs.
- The IP GT (Gateway Table) contains routing information to reach specified remote networks.

Note Do not configure IP information for 91750 nodes or HP 3000 nodes that do not support NS Common Services or ARPA Services.

Router/1000 NRV Information

The information in the Router/1000 Nodal Routing Vector is used for two purposes:

- network routing on Router/1000 Directly-Connected Networks
- address information for DS/1000-IV Compatible Services (RTE-RTE).

Therefore, you must make an entry in the NRV for a remote node if

- the remote node and the local node are members of the same Router/1000 network

OR

- the remote node supports DS/1000-IV Compatible Services (RTE-RTE).

Worksheets

This subsection uses the internet in Figure 3-11 as an example for configuring routing information.

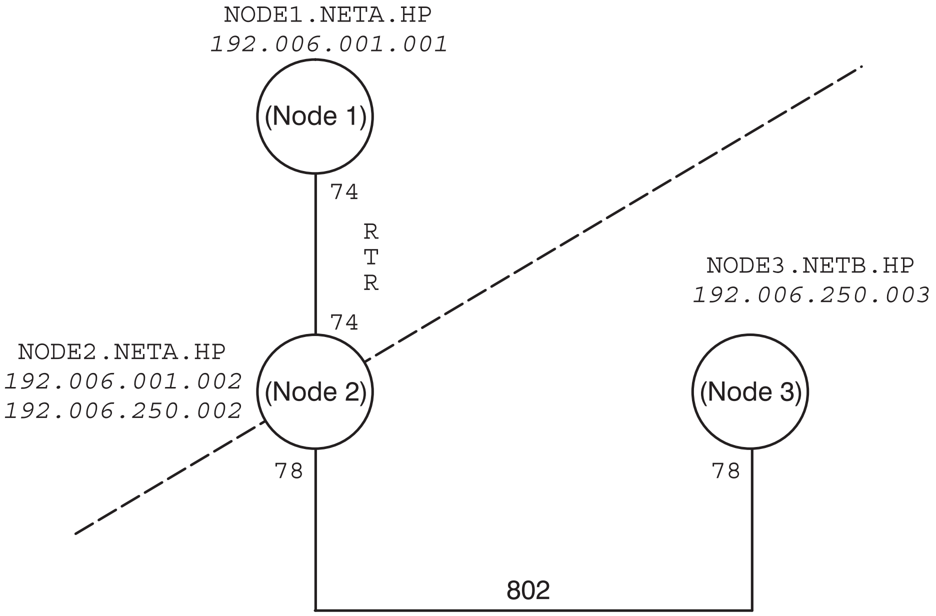


Figure 3-11. Sample Network Map with Names and Addresses

In Figure 3-11, there are three nodes and two networks linked by the LI types, 802 and RTR, noted in the diagram. Dotted lines indicate the network boundaries.

Figure 3-12 shows the routing worksheet for Figure 3-11. NODE2.NETA.HP is a gateway; it belongs to two networks and has two IP addresses (*192.006.001.002*, for its RTR link, and *192.006.250.002*, for its 802 link).

All the IP addresses are Class C addresses; the first nine digits of each address indicate the network address, and the last three digits indicate the node address (for example, the network address for *192.006.001.002* is *192.006.001*, and the node address is *002*).

The numbers next to the LI types indicate the transmit LUs. In this example, both machines on the opposite ends of a given link use the same transmit LU. For example, NODE1.NETA.HP and NODE2.NETA.HP each use LU 74 for the RTR link's transmit LU. Note that this example is not an example for the sample system generation answer file, /NS1000/EXAMPLES/#ANSNS or for the RTE-A primary system generation answer file. The LAN LU in those answer files is LU 96.

Node Name IP Address(es)	NODE1.NETA.HP *192.006.001.001*	NODE2.NETA.HP *192.006.001.002* *192.006.250.002*	NODE3.NETB.HP *192.006.250.003*
Router/1000 Address	1	2	3
DCN Information	*192.006.001.001*,	*192.006.001.002*,	*192.006.001.003*,
local IP addr,	RTR	RTR	802,,78
RTR or LAN,		*192.006.250.002*,	
segsize,		802,,78	
If LAN:			
link lu, E or NE,			
station address	/E	/E	/E
GT Information	*192.006.250.002*,	/E	*192.006.001.001*,
Dest IP Net,	*192.006.001.002*		*192.006.250.002*
Gateway IP, Hops			
	/E		/E
NRV Entries	2, *192.006.001.002*,	1, *192.006.001.001*,	1, *192.006.001.001*,
node number,	,,,,MA	,,,,MA	,,,,MA
IP Address,	3, *192.006.250.003*,	3, *192.006.250.003*,	2, *192.006.250.002*,
LU, LU T/O,Level,	,,,,MA	,,,,MA	,,,,MA
N, MA, *MA T/O*			

Figure 3-12. Sample Routing Worksheet

In the routing worksheet, you can leave a field blank, which indicates that NSINIT is to use the default value for that field. If you specify any subsequent fields, you must enter a comma placeholder.

The Gateway Table entry for NODE1.NETA.HP is *196.006.250.002*, *192.006.001.002*. This entry indicates to IP to send packets destined for the *192.006.250.xxx* networks to the gateway node, *196.006.001.002*.

The Gateway Table can include a destination network address of *000.000.000.000* for use with a default gateway. Refer to the subsection "IP Gateway Table (GT)" later in this section for more details.

IP Directly Connected Network (DCN)

IP uses the DCN information to determine which Router/1000 or LAN networks the local node belongs to and to determine the local IP addresses for a given node. Make a DCN entry (Section 9 of the Network Worksheet) for each Router/1000 network or LAN network to which the local node is directly connected. If the local node has no RTR or 802/Ethernet/LAN LIs, leave the DCN section of the Network Worksheet blank, or enter `/E` if you are using the Routing Worksheet.

Up to three LAN cards are supported in an NS-ARPA/1000 node, which means up to three DCN entries allowed for LAN. For example, in Figure 3-12, NODE1.NETA.HP has one RTR link, so it has one DCN entry. In the column for NODE2.NETA.HP, there are two DCN entries because that node has a RTR and an 802 LI and is a gateway.

Parameters

local IP addr The local node's IP address associated with the RTR or 802/Ethernet/LAN LI.

[*subnet mask*] The subnet mask used for subnetted networks. It is 32 bits, with bits set to 0 for the node number and 1 for the network address and subnet number.

The same subnet mask should be used for all subnetworks in the network. Subnet masks should be entered in decimal notation; e.g., 255.255.255.0.

The first subnet mask entered becomes the default subnet mask for subsequent IP addresses. For a non-subnetted network, the default subnet mask is as follows:

Class A: 255.0.0.0
Class B: 255.255.0.0
Class C: 255.255.255.0

`RTR` Indicates that the IP address is associated with a Router/1000 network.

`802` Indicates that the IP address is associated with an IEEE 802.3 LI.

`ETHERNET` Indicates that the IP address is associated with an Ethernet LI.

`LAN` Indicates that the IP address is associated with both IEEE 802.3 and Ethernet LIs.

segsize The network segment size; the largest size buffer (in bytes) that the link level protocol will accept. *HP recommends that you use the the default network segment size*. For more information, refer to the "NSINIT" section, IP subsection.

Default: 1514 bytes.

link lu *(For 802/Ethernet/LAN LIs only)*. The transmit (user) LU of the LAN interface card for the network indicated by *local IP addr*.

`E`
`NE`

(For 802/Ethernet/LAN LIs only). `E` specifies that you want NSINIT to enable the link at initialization time; `NE` indicates that you do not want to enable the link.

Default: `E`.

`station address`

(For 802/Ethernet/LAN LIs only). The LAN card's station address. If you leave this field blank, NS-ARPA uses the card's station address (the address in the card's NOVRAM).

If using the Routing Worksheet, enter `/E` when you have finished entering the DCN information for a node.

IP Gateway Table (GT)

IP uses the Gateway Table (GT) to determine which gateway to route through to reach a remote network. If you have only one network (you do not have an internet), leave the GT section (Section 10) blank, or enter `/E` if you are using the Routing Worksheet. Otherwise, make a GT entry for each remaining remote network in the internet that you have not listed in the DCN for the local node. The remaining remote networks are those that are not directly connected to the node you are configuring or connected via a gateway link.

For example, in Figure 3-11 and Figure 3-12, NODE1.NETA.HP has one GT entry.

The GT entries contain the following fields:

Parameters

`Dest IP Net`

The IP address of any node on the remote network. It can be the address of any node, because IP uses only the network portion of the address. If the remote node has more than one IP address, this must be the IP address associated with the remote network for which you are making an entry.

The IP address can also be *`000.000.000.000`* to indicate any remote network that is not defined in the GT. Messages for a remote network that is not defined in the GT are sent to the associated gateway node, *`Gateway IP`*. This gateway node is called a default gateway.

`Gateway IP`

The IP address of the node through which all messages for the specified remote network are sent. This node is the gateway to other networks. This node must be on the same network as the local node. The network address portion of the Gateway IP must match the network address portion of an entry in the DCN table.

In Figure 3-12, the GT entry for NODE3.NETB.HP indicates that IP would route messages destined for network *`192.006.001`* through the gateway *`192.006.250.002`*. Note that the gateway's network address matches the network portion of the IP address for NODE3.NETB.HP (*`192.006.250`*).

If `Dest IP Net` has an IP address of `000.000.000.000`, then the IP address given for `Gateway IP` will be the default gateway. The default gateway will send messages to remote networks that are not specified in the Gateway Table.

Hops The maximum number of gateways that IP is to route through before reaching the remote network. *HP recommends that you use the default value*. For more information, refer to the "NSINIT" section, IP subsection.

Default: 100

If you are using the Routing Worksheet, enter `/E` when you have finished entering the GT information for a node.

If you want to specify a default gateway, the GT portion of Figure 3-11 and Figure 3-12 can be rewritten as shown in Figure 3-13 and Figure 3-14. Using a default gateway simplifies the Gateway Table. Your node only needs to know the IP address of your default gateway to reach any node in the internet.

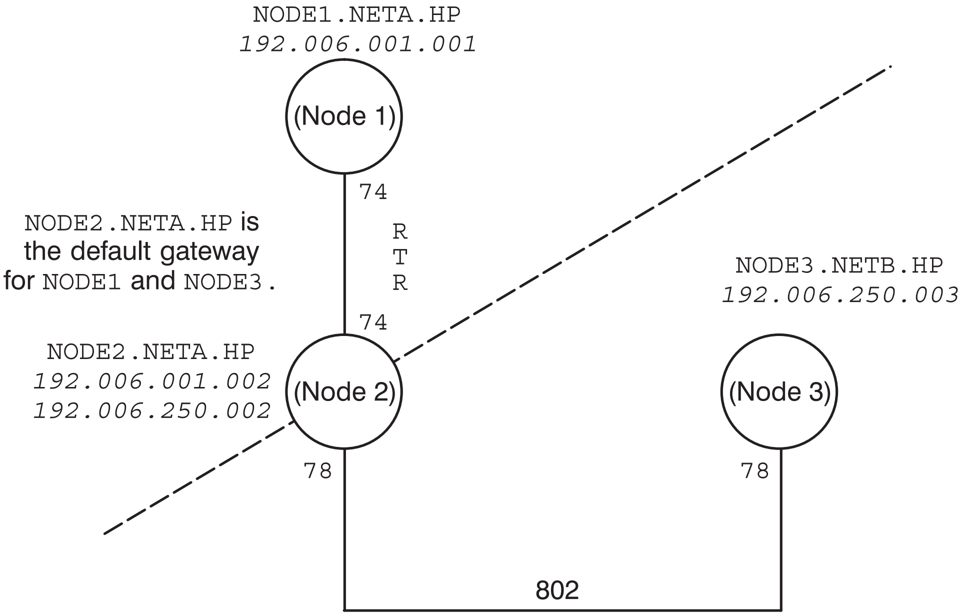


Figure 3-13. Node 2 is the Default Gateway for Nodes 1 and 3

Node Name IP Address(es)	NODE1.NETA.HP *192.006.001.001*	NODE2.NETA.HP *192.006.001.002* *192.006.250.002*	NODE3.NETB.HP *192.006.250.003*
Router/1000 Address	1	2	3
DCN Information *local IP addr*, RTR or LAN, *segsize*, If LAN: *link lu*, E or NE, *station address*	*192.006.001.001*, RTR	*192.006.001.002*, RTR *192.006.250.002*, 802,,78	*192.006.001.003*, 802,,78
	/E	/E	/E
	default gateway specification:		default gateway specification:
GT Information *Dest IP Net*, *Gateway IP, Hops*	*000.000.000.000*, *192.006.001.002*	/E	*000.000.000.000*, *192.006.250.002*
	/E		/E
NRV Entries *node number*, *IP Address*, *LU, LU T/O*,Level, N, MA, *MA T/O*	2, *192.006.001.002*, ,,,,MA 3, *192.006.250.003*, ,,,,MA	1, *192.006.001.001*, ,,,,MA 3, *192.006.250.003*, ,,,,MA	1, *192.006.001.001*, ,,,,MA 2, *192.006.250.002*, ,,,,MA

Figure 3-14. Sample Routing Worksheet with Dest IP Net Address of Zero for Figure 3-13

Router/1000 Nodal Routing Vector (NRV)

Router/1000 uses the NRV for subnet routing within Router/1000 networks and address information for DS/1000-IV Compatible Services (RTE-RTE). Make an entry in the NRV for the following nodes:

- Make an entry for every remote HP 1000 on the local network connected via a Router/1000 link.

- Make an entry for every remote HP 1000 *in the internet* that supports DS/1000-IV Compatible Services, including DS/1000-IV (91750) nodes.

Fill out the NRV information for each node being entered as follows:

- If the node has one IP address, use it.

- If the node is a member of more than one network and one of the networks is a local network, use the IP address for the local network.

- If the node is a member of more than one network and none of the networks is a local network, use any IP address for this node.

- If the node has no IP address (because it is a DS/1000-IV node) and the local node is a member of the same network, do not use an IP address.
- If the node has no IP address (because it is a DS/1000-IV node) and the local node is not on the directly connected network (DCN), assign a guardian node IP address.

Parameters

node number — The Router/1000 address of the remote node. This must be an integer between 1 and 32767, inclusive. Address 0 (zero) is not allowed.

IP Address — The remote node's IP address. If the remote node is an NS-ARPA/1000 node and is a member of a Router/1000 network, *this must be the IP address associated with the Router/1000 network*. For example, in Figure 3-11 and Figure 3-12, all of the NRV entries that specify NODE2.NETA.HP contain its IP address associated with the Router/1000 network, *192.006.001.002.*

If the remote node is not a member of a Router/1000 network and has more than one IP address, you can enter any one of its IP addresses.

For 91750 nodes, enter the IP address associated with the RTR LI of the guardian 91790 node that the local node is to use to reach the 91750 node. However, if the local node and the 91750 node belong to the same Directly Connected Network (DCN), do not assign a guardian node; leave this field blank. For general information on guardian nodes, refer to "Guardian Nodes for DS/1000-IV (91750) Nodes" earlier in this section.

LU — The transmit LU of the link interface card to the remote node. Enter the transmit LU here only if the remote node and the local node are members of the same Router/1000 network and NS-ARPA is *not* to use Dynamic Rerouting to the remote node.

Leave this field blank if: the remote node and the local node are not members of the same Router/1000 network; or if NS-ARPA is to use Dynamic Rerouting to the remote node. HP recommends that you use Dynamic Rerouting over HDLC RTR links.

LU T/O — The transaction timeout in seconds. This is the maximum number of seconds the local system will wait for a reply from the remote system before reporting a timeout error. You should have the same link timeout for the local and remote system.

Default: the Master timeout value (configured in the DS/1000-IV Compatible Services section of the NSINIT dialogue).

Level — The software level. Enter `1` for DS/1000-IV (91750) and NS-ARPA/1000 (91790) nodes.

Default: 1.

`N` Indicates that the remote node is a neighbor (there is a direct link to this node). If Dynamic Rerouting is used (you did not specify an LU for this node), this field is optional.

`MA` Indicates that NS-ARPA is to use Message Accounting for transactions with this node.

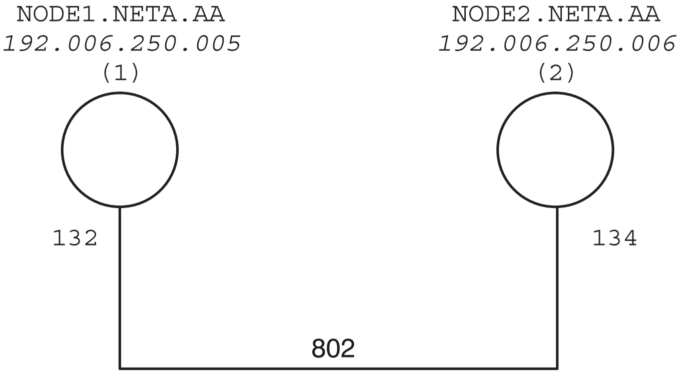
Default: No MA.

`MA T/O` The Message Accounting timeout value for transactions with the remote node. This is the number of seconds that Message Accounting will wait for an acknowledgment to arrive for a message before timing out. MA will then retransmit (retry) the message. MA will retransmit a message as many times as the MA Retry count. The MA timeout value times the MA retry limit must be less than or equal to the Master timeout value (configured in the DS/1000-IV Compatible Services section of the NSINIT dialogue).

Default: 3.

NS-ARPA RTR links default to have rerouting. DS/1000-IV links default to non-rerouting. For the “Router/1000 Links” subsection of the NSINIT dialogue (refer to Section 9 of this manual) if you specify the default, NSINIT will override the placement of the LU in the NRV and the link will be non-rerouting.

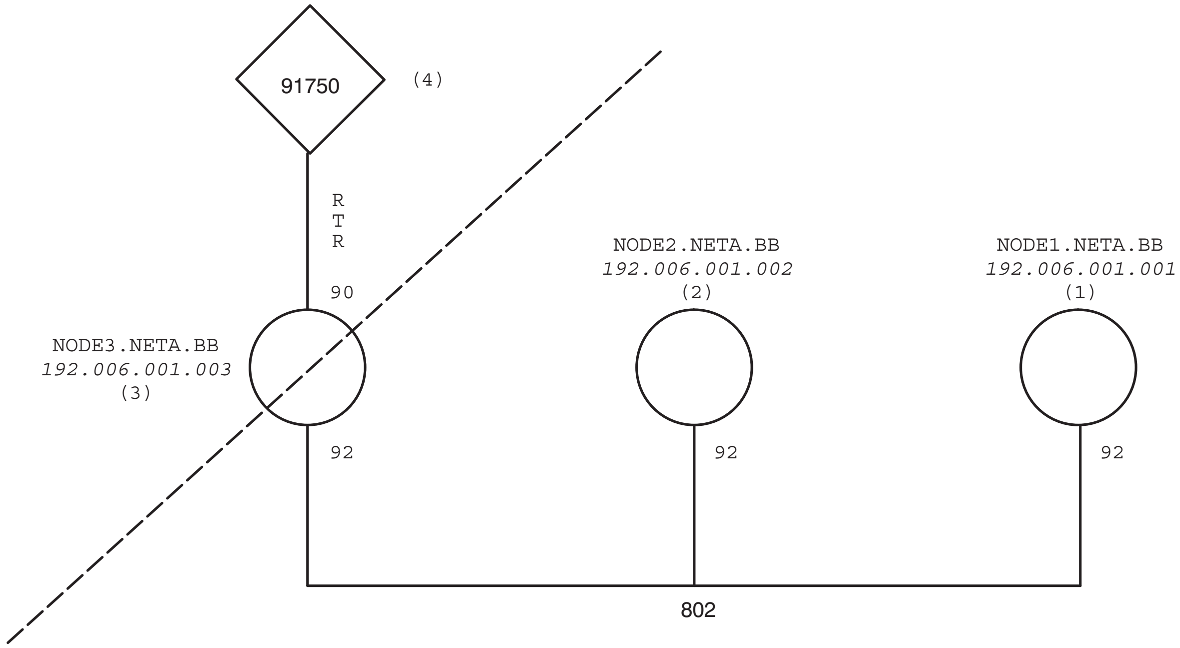
Routing Worksheet Example for IEEE 802.3 Network



Node Name IP Address(es)	NODE1.NETA.AA *192.006.250.005*	NODE2.NETA.AA *192.006.250.006*
Router/1000 Address	1	2
DCN Information *local IP addr*, RTR or LAN, *segsize*, If LAN: *link lu*, E or NE, *station address*	*192.006.250.005* 802,,132 /E	*192.006.250.006* 802,,134 /E
GT Information *Dest IP Net*, *Gateway IP, Hops*	/E	/E
NRV Entries *node number*, *IP Address*, *LU, LU T/O,Level*, N, MA, *MA T/O*	2, *192.006.250.006*, ,,,,MA	1, *192.006.250.005*, ,,,,MA

Figure 3-15. Network Map and Routing Worksheet for an IEEE 802.3 Network

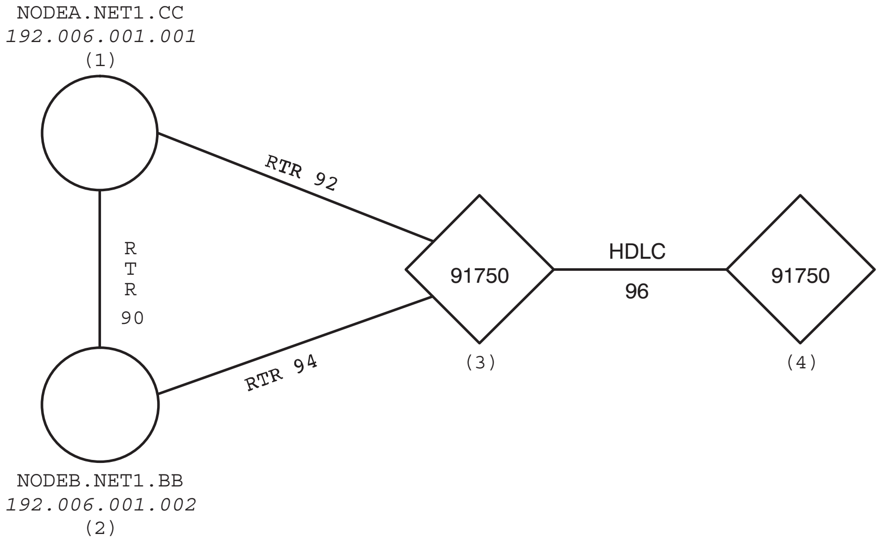
Routing Worksheet Example for IEEE 802.3 Network and DS/1000-IV Node



Node Name IP Address(es)	NODE1.NETA.BB *192.006.001.001*	NODE2.NETA.BB *192.006.001.002*	NODE3.NETA.BB *192.006.001.003* *192.006.250.003*	<NONE> <NONE>
Router/1000 Address	1	2	3	4
DCN Information *local IP addr*, RTR or LAN, *segsize*, If LAN: *link lu*, E or NE, *station address*	*192.006.001.001*, 802,,92 /E	*192.006.001.002*, 802,,92 /E	*192.006.001.003*, 802,,92 192.006.250.003, RTR /E	<NONE>
GT Information *Dest IP Net*, *Gateway IP*, *Hops*	/E	/E	/E	<NONE>
NRV Entries	/E	/E	/E	
node number, *IP Address*, *LU, LU T/O*,*Level*, N, MA, *MA T/O*	2, *192.006.001.002*, ,,,,MA 3, *192.006.001.003*, ,,,,MA 4, *192.006.001.003*, ,,,,MA	1, *192.006.001.001*, ,,,,MA 3, *192.006.001.003*, ,,,,MA 4, *192.006.001.003*, ,,,,MA	1, *192.006.001.001*, ,,,,MA 2, *192.006.001.002*, ,,,,MA 4,,,,,,MA	1 2 3

Figure 3-16. Network Map and Routing Worksheet for IEEE 802.3 Network and DS/1000-IV Node

Routing Worksheet Example for RTR Network and DS/1000-IV Node



Node Name IP Address(es)	NODEA.NET1.CC *192.006.001.001*	NODEB.NET1.BB *192.006.001.002*	<NONE> <NONE>	<NONE> <NONE>
Router/1000 Address	1	2	3	4
DCN Information *local IP addr*, RTR or LAN, *segsize*, If LAN: *link lu*, E or NE, *station address*	*192.006.001.001*, RTR /E	*192.006.001.002*, RTR /E	<NONE>	<NONE>
GT Information *Dest IP Net*, *Gateway IP, Hops*	/E	/E	<NONE>	<NONE>
NRV Entries *node number*, *IP Address*, *LU, LU T/O,Level*, N, MA, *MA T/O*	2, *192.006.001.002*, ,,,,MA 3,,,,,,MA 4,,,,,,MA	1, *192.006.001.001*, ,,,,MA 3,,,,,,MA 4,,,,,,MA	1 2 4	1 2 3

Figure 3-17. Network Map and Routing Worksheet for Router/1000 Network DS/1000-IV Nodes

Security Codes

There are three types of security codes that you can assign:

- Network Security Code
- Network User's Security Code
- Bisync ID Sequences (Dial-up Bisync links used for DS/1000-IV Compatible Services [RTE-MPE] only).

Network Security Code

The Network security code consists of two alphanumeric characters and is required for the following tasks:

- shutting down NS-ARPA (via NSINIT)
- establishing Remote I/O maps
- modifying timing parameters and changing the NRV (via DSMOD)

Network User's Security Code

The Network User's security code consists of two alphanumeric characters. It is required to execute the REMAT `SW` (SWITCH) command; thus you can use it to restrict REMAT access to remote nodes.

Bisync ID Sequences

If you have an HP 3000 connected to an HP 1000 via a Bisync dial-up link that supports DS/1000-IV Compatible Services, you can assign local and remote ID sequences to limit access to the HP 1000. When the HP 1000 or an HP 3000 attempts to establish communication over the link, each machine sends its local ID sequence, which the other machine compares with its remote ID sequence.

ID sequences can be 1 to 15 characters long.

Worksheet Entry

Enter the Bisync ID sequences in Section 13 of the Network Worksheet.

4

Internal Resources

This section describes the following resources required for NS-ARPA/1000:

- Disk Space
- Memory
- Software modules
- System resources

You will need this information when generating and booting up your system, installing software modules, and initializing your node.

At the end of the subsections for the memory areas, software modules, and system resources is a summary and set of guidelines for allocating and selecting these resources. In most cases, the information in the summary and guidelines is all you need to know to configure your internetwork.

Disk Space

NS-ARPA/1000 uses disk space for product files and installed files. Product files are those files on the product tape. Installed files are those files created during the installation process and remain on the system.

The standard disk space terminology is as follows:

```
1   block  = 256 bytes
1   Mbyte  = 1,000,000 bytes
           = 3906.25 blocks
```

NS-ARPA/1000 uses the disk space as follows:

```
/NS1000/@.@.S = 20,000 blocks (5.12 Mbytes) for the product files on the product tape.

/PROGRAMS     = 13,000 blocks
/LIBRARIES    =  3,800 blocks
/SYSTEM       =    250 blocks
/HELP         =    180 blocks
/INCLUDE      =    170 blocks
/ETC          =     10 blocks
/CATALOGS     =     20 blocks
              = 17,430 blocks (4.46 Mbytes) total for installed files.
```

During the installation process, disk space is needed for both product and installed files which uses about 10.5 Mbytes of disk space (see calculations below). The product files can be removed from the disk after the installation is complete.

```
Product files      = 20,000 blocks (5.12 Mbytes)
Installed files    = 17,430 blocks (4.46 Mbytes)
Installation total = 37,430 blocks (9.58 Mbytes)
```

Memory Areas

NS-ARPA/1000 uses five memory areas:

- Distributed System Available Memory (DSAM)
- System Available Memory (SAM)
- Extended System Available Memory (XSAM)
- System Memory Block (SMB)

Table 4-1 lists the system defaults used for NS-ARPA/1000 for each of the resources described in this section. SMB guidelines are described in Table 4-2 later in this section.

Table 4-1. NS-ARPA/1000 Default for Internal Resources

Resource	Default (Recommended) Allocation
DSAM	151 pages
SAM	32 pages*
XSAM	(50 words × # of concurrent TELNET programs) + 2 pages

* 32 pages of SAM are recommended; however, NS-ARPA can work with fewer pages of SAM.

Distributed System Available Memory (DSAM)

DSAM is memory area specifically reserved for NS-ARPA/1000. Most NS-ARPA/1000 tables are stored in DSAM. In addition, DSAM is used to buffer messages and to pass messages between programs.

DSAM is controlled by the NS-ARPA/1000 Memory Manager, and is implemented as a Sharable Extended Memory Access (SHEMA) partition, `#DSAM`. `#DSAM` is initially allocated by MMINIT (Memory Manager initialization program). MMINIT renames `#DSAM` to `,,,DS` when NSINIT schedules it.

This subsection describes the following tasks:

- DSAM should be in a reserved partition.
- DSAM size selection. NS-ARPA/1000 provides an installation command file, `/NS1000/CMD/INSTALL_NS1000.CMD` , to load NS-ARPA/1000 programs, including MMINIT. In most cases, the size of the DSAM SHEMA partition specified in this command file is more than adequate, and you do not have to alter this value.

DSAM Reserved Partition

Place `#DSAM` in a reserved partition. Placing `#DSAM` in a partition is described in the section "BOOT and WELCOME Files." `#DSAM` should be assigned to a reserved partition for the following reasons:

- `#DSAM`, like all SHEMAs, is never swapped out of memory.
- When `#DSAM` is in a reserved partition and there is a parity error in its memory area, subsystem shut down is more graceful, and the state that NS-ARPA was in at the time of the parity error is examinable.

To assign `#DSAM` to a reserved partition, you must define a reserved partition at boot time. In most cases, you can use the partition size set up by the NS-ARPA/1000 file `MMINIT.LOD`. The following subsections provide more information on determining DSAM size requirements.

DSAM Size

The Sharable EMA size specified for MMINIT in the file `MMINIT.LOD` is 150 pages. RTE allocates a partition that is one page larger than that, so if you use the specification in the NS-ARPA/1000 load file, DSAM will be 151 pages. This is more than adequate for most nodes. If you have no reason to increase or decrease this size, you can skip the rest of this discussion.

When you initialize a node with NSINIT, NSINIT prints the amount of DSAM required for NS-ARPA tables. In addition, you must allocate DSAM for message buffers.

There are many factors that determine DSAM requirements, and some of them may vary from one run of a program to the next. Like SAM, the memory buffers in DSAM are allocated dynamically when needed. The number of buffers needed will depend on network activity. Therefore, there is no exact formula for determining DSAM requirements. Some guidelines are given in the following subsections.

Decreasing DSAM

You may decrease the size of DSAM if networking activity on your system is limited.

The NSINF `C` command can be used to display the number of DSAM buffer memory pages currently in use.

You should not set the size of DSAM smaller than the size required for tables (reported by NSINIT) plus the number of buffer pages typically in use, plus an additional 32 pages.

NSINF is documented in the *NS-ARPA/1000 Maintenance and Principles of Operation Manual*.

Increasing DSAM

You may want to increase the size of DSAM under the following circumstances:

- NetIPC programs fail or suspend when they try to create sockets (via `IPCCreate`).
- You get error or logging messages that indicate there is not enough DSAM to complete a task.
- The Memory Manager statistic `Callers Waited`, displayed by NSINF, indicates that several applications had to wait for DSAM.

NS Common Services and ARPA Services

For NS Common Services and ARPA Services, there are two factors that determine the amount by which you increase DSAM:

- the number of additional NS-ARPA (NetIPC, NFT, RPM, FTP, or TELNET) programs you want to support
- the size of messages that NetIPC and/or RPM programs send

For the number of additional programs that you want to support, use the number of programs that you see suspended or failing at any time due to lack of memory. Only count programs that are suspended or failing at any one time, since programs that run at different times do not compete for the same resources. Add one page to DSAM for each additional program that you want to support.

For the size of the messages that these programs will send, use the the `IPCConnect` or `IPCRecvCn maximum send size` option for outbound messages and the `IPCConnect` or `IPCRecvCn maximum receive size` options for inbound messages. The default for these options is 100 bytes. For each 600 bytes that a program sends or receives, add approximately one page to DSAM.

NFT

For each NFT transaction there is an *Initiator* node, *Producer* node, and *Consumer* node. The Initiator node is the node where the NFT request originates. The Producer node is the the node at which the file to be copied exists. The Consumer node is the node to which the copied file will be sent.

When two NFT monitors negotiate for a connection, they exchange NFT default buffer sizes. The actual buffer size used is the lower of the two default buffer sizes; however, DSAM requirements are based on the configured value.

Add approximately one page to DSAM for every 1400 bytes in the NFT buffer at each node (Initiator, Producer, and Consumer). If a node provides more than one function in a transaction, then the node requires the sum of those function's requirements. For example, if node is the Initiator and the Producer node, then that node requires approximately `2 * (NFT Buffer Size/1400)` pages.

Configuration: The default buffer size is configured in the NFT section of the NSINIT dialogue. The default is 2048 bytes.

RPM

RPM needs at least two pages of DSAM. In addition, each child or parent program needs 1.5 pages of DSAM.

TELNET

For each TELNET server, one page of DSAM is needed. In addition, for each TELNET user program, four pages of DSAM is needed.

FTP

For each FTP user program or FTP server, seven pages of DSAM is needed.

DS/1000-IV Compatible Service Messages (RTE-RTE)

Messages for DS/1000-IV Compatible Services (RTE-RTE) that will be transmitted over a non-Router/1000 link pass through SAM *and DSAM* before they are transmitted. For messages of this type, determine the length of the requests and replies plus any data and the attached DS/1000-IV headers and RTE system headers and add approximately one page to DSAM for every 600 bytes. The longest DS/1000-IV (RTE-RTE) message header is 74 bytes (PTOP calls).

Summary and Guidelines

Reserve a partition for `#DSAM` in your boot command file. Link or relink MMINIT to set the #DSAM SHEMA size and assign it to the partition reserved for it.

If you need more DSAM, add one page for each additional NS-ARPA program, and approximately one page for every 600 bytes configured for the inbound and outbound message buffers.

System Available Memory (SAM)

All inbound and outbound NS-ARPA messages pass through SAM. SAM is used as a buffer storage area between the NS-ARPA device drivers and the Transport. If a message is received or a request is made and there is not enough SAM to store it in, the transaction may be delayed and performance may degrade.

There is no exact formula for determining SAM requirements. If you do not have enough SAM, however, there will be obvious signs of system performance degradation.

One way to allocate SAM is to initially configure the maximum amount of SAM (32 pages). You can then run your applications and decrease the SAM's size so long as system performance is acceptable as described in the following subsections.

For systems that do not have 32 pages of SAM, the following subsections describe other ways to determine SAM allocation. Otherwise, allocate 32 pages for SAM and skip the remainder of this discussion.

SAM Allocation

In most cases, it is best to initially allocate 32 pages of SAM. However, if a system cannot support 32 pages of SAM you can allocate SAM using the following guidelines:

- If you had a working system with DS/1000-IV (91750), the current SAM allocation is probably adequate.
- If you have a working system with neither NS-ARPA/1000 nor DS/1000-IV, add about five pages to the current SAM allocation.

If you do not have a working system, use the following guidelines for SAM allocation:

- Allocate about 200 words for each concurrently active user session you expect to have. (This is for file system UDSP tables and class I/O read buffers for CI prompts.)
- If you only have LAN/1000 links, add two pages for NS-ARPA/1000 message buffering. Otherwise, add five pages for NS-ARPA/1000 message buffering.
- Take the sum of the user requirement and the link requirement and increase it by one-third (to allow for SAM fragmentation).
- Add about one more page of SAM for each of the following conditions on your system:
    - several concurrently active NS-ARPA/1000 applications (includes using the NS Common Services and ARPA Services—TELNET, RPM, NetIPC, and so forth)
    - frequent use of NS-ARPA/1000 message tracing (NSTRC)
    - several Router/1000 links with dynamic rerouting.

Each TELNET user program uses two pages of SAM.

Performance Degradation and SAM Size

The following situations indicate performance degradation caused by an inadequate SAM size:

- `DS08(0)` errors reported by DS/1000-IV Compatible Services, or remote busy messages reported by Transparent File Access (TRFAS) may indicate that the remote node does not have enough SAM.
- NS Common Services and ARPA Services programs are suspended, waiting for SAM (as displayed by the `WH` command).
- Poor throughput for NS Common Services and ARPA Services programs may also indicate that a node does not have enough SAM.

Summary and Guidelines

Because all messages must be stored in SAM, *you may want to allocate about one-third more SAM than you expect your applications to use*. Allocating extra SAM allows for memory fragmentation, which is a consideration particularly if you have large NetIPC or PTOP data buffers. Remember that 32 pages is the maximum for SAM.

When calculating SAM requirements, you may be able to separate the programs into functional groups that do not run concurrently. If so, you can calculate the SAM requirements for each such group; the SAM you require will be the maximum value you calculate for any one group.

Extended System Available Memory (XSAM)

Extended SAM is used as additional buffer storage area. The XSAM requirements for NS-ARPA are as follows:

- TELNET uses the RTE signal mechanism to facilitate the mapping of terminal I/O to NS-ARPA socket events.

  Each TELNET user program needs about 50 words of XSAM. Thus, the TELNET XSAM requirement can be determined by the following formula:

  ```
  Additional XSAM = Number of concurrent TELNET programs (User &
  Server) x 50 words
  ```

- About 2 pages of additional XSAM should be allocated to avoid “starvation”. XSAM is a shared resource among all system users. The extra XSAM should alleviate contention for it among the users.
- TELNET and NFT use the shared program feature of RTE-A. Increase XSAM by 50 words for each TELNET and NFT program running concurrently.

System Memory Block (SMB)

SMB is memory area in the system map that is specified at system generation time by the generator `MB` command. SMB is used for DS/1000-IV Compatible Services tables for transaction and user information. When you initialize a node, NSINIT prints the number of words required for the SMB according to configuration values you specify. SMB requirements are listed in the following subsections, and are summarized in Table 4-2.

If you do not have DS/1000-IV Compatible Services generated into the system, then you need at least one word of SMB.

If you configure the *default* quantities for DS/1000-IV, you will need *300* words for the SMB.

If you configure the *maximum* quantities for all of the items below, you will need *3312* words for the SMB.

DS/1000-IV Transactions

Each programmatic or remote request (such as REMAT, RMOTE, PTOP, HELLO, Remote I/O Mapping) that uses DS/1000-IV Compatible Services is a DS/1000-IV transaction. Each DS/1000-IV transaction requires a transaction control block (TCB). TCBs used for transactions to HP 3000s are entered in the Process Number List (PNL).

Each TCB requires 6 words in SMB.

Configuration: The maximum number of active DS/1000-IV transactions is configured in the DS/1000-IV Compatible Services section of the NSINIT dialogue. The default is 20.

DS/1000-IV Remote Access (RTE-RTE)

Each concurrent PTOPM or EXECW schedule request from a remote node to the local node uses an entry in the POOL table. The number of entries created in POOL is the maximum number of DS/1000-IV remote transactions configured via NSINIT.

Resources: The POOL table header requires 1 word in SMB; each POOL entry requires 7 words in SMB.

Configuration: The maximum number of concurrent DS/1000-IV remote transactions is configured in the DS/1000-IV Compatible Services section of the NSINIT dialogue. The default is 7.

DS/1000-IV Remote Users (RTE-MPE)

Each concurrent NS/3000 or DS/3000 request that accesses this node to use DS/1000-IV Compatible Services (RTE-MPE) requires an entry in the Transaction Status Table (TST).

Resources: Each TST entry requires 14 words in the SMB.

Configuration: The maximum number of concurrent DS/1000-IV remote users (RTE-MPE) is configured in the DS/1000-IV Compatible Services section of the NSINIT dialogue. The default is 4.

LUs for DS/1000-IV Compatible Services (RTE-MPE)

Each PSI/Bisync link and X.25 pool LU used for DS/1000-IV Compatible Services (1000-3000) requires a DS/1000-IV—DS/3000 LU table entry.

Resources: Each table entry requires 2 words in SMB.

Configuration: The number of X.25 pool LUs and the PSI/Bisync LUs used for DS/1000-IV Compatible Services (RTE-MPE) are configured in the DS/1000-IV Compatible Transport section of the NSINIT dialogue.

Summary and Guidelines

Table 4-2 lists guidelines for SMB allocation.

Table 4-2. SMB Guidelines

Activity	Item	Amount of Memory	Comments
DS/1000-IV transactions	Transaction Control Block (TCB)	6 words	Each program or remote access that uses DS/1000-IV Compatible Services requires a TCB. The default number of TCBs is 20; the maximum number is 200.
Remote Access (RTE-RTE)	POOL table entries	1 word table header plus 7 words per entry	Each concurrent program scheduled by a PTOPM or EXECW request from a remote node at the local node uses an entry in the POOL table. The default number of POOL entries is 7; the maximum number allowed is 253.
Remote Users (RTE-MPE)	Transaction Status Table (TST) entries	14 words	Each remote user that accesses the local node to use DS/1000-IV Compatible Services (RTE-MPE) requires a TST entry. The default number of TST entries is 4; the maximum number allowed is 10.
LUs for DS/1000-IV Compatible Services (RTE-MPE)	DS/1000-IV—DS/3000 LU table entries	2 words	Each X.25 pool LU and Bisync LU used for DS/1000-IV Compatible Services (RTE-MPE) requires an entry. The default number of entries is 0; the maximum number entries allowed is 100.

If you configure the *default* quantities for all of the above items, you will need *300* words for the SMB. If you configure the *maximum* quantities for all of the above items, you will need *3312* words for the SMB.

Software Modules

The following modules are used for NS-ARPA/1000. Some modules are required for certain configurations; other modules are optional, depending on the features and user services offered.

Some modules are *monitors*. When scheduled (at node initialization time or by DSMOD), each Monitor is allocated a NetIPC Call Socket or assigned an RTE class number. The monitors remain scheduled until NS-ARPA is shut down; those that are allocated a Call Socket wait for an `IPCRecvCn` on the socket, and those assigned a class number wait for completion of a class get. There are three types of monitors:

- *Transport Monitors* process inbound or outbound messages. They act as an interface between the user services and the communication device drivers.
- *Slave Monitors* service incoming requests for local resources from remote nodes.
- *Converters* are used to convert message formats for DS/3000 and DS/1000 services. If Converters are needed, they are always scheduled, waiting for a class get completion. In addition, there are *watch dogs*, which are scheduled at regular time intervals and clean up resources.

The following subsections describe the modules required for specific NS-ARPA activities. You must load the modules required for your configuration before running NSINIT to initialize NS-ARPA. To load the NS-ARPA modules, you can use the NS-ARPA/1000 file `/NS1000/CMD/INSTALL_NS1000.CMD` , as described in Section 6, "NS-ARPA/1000 Program and File Installation." In addition, you must restore (`RP`) some of the programs. The list at the end of this section summarizes which modules are required for different configurations, and which programs must be restored.

The modules are divided into the following groups and a summary is given at the end of this section in "NS-ARPA Summary and Guidelines":

- *Required in All Nodes*—modules that are required in all NS-ARPA/1000 nodes, regardless of the services used.
- *Recommended for All Nodes*—HP recommends that you load these modules in all nodes, regardless of the services used.
- *Related Subsystem Modules*—modules from related subsystems (RTE Transparent File Access, Image/1000) that you must relink to be with NS-ARPA/1000.
- *Required for Specific Links*—modules that are required for specific links, regardless of the services used.
- *Required for Specific NS Features and Services*—modules that are required only if you want specific NS features and services.
- *Required for Specific ARPA Services*—modules that are required only if you want specific ARPA services.
- *Required for DS/1000-IV Compatible Services (RTE-RTE)*—modules that are required if you want DS/1000-IV Compatible Services (RTE-RTE), regardless of the specific services used.

- *Required for Specific DS/1000-IV Compatible Services (RTE-RTE)*—modules that are required only if you want specific DS/1000-IV Compatible Services (RTE-RTE).
- *Required for DS/1000-IV Compatible Services (RTE-MPE)*—modules that are required if you want DS/1000-IV Compatible Services (RTE-MPE), regardless of the specific services used.
- *Required for Specific DS/1000-IV Compatible Services (RTE-MPE)*—modules that are required only if you want specific DS/1000-IV Compatible Services (RTE-MPE).

The module descriptions note any system resources required for a given module; additionally requirements for each system resource are summarized in the subsection “System Resources.”

Required in All Nodes

The following modules are required in all nodes regardless of services used:

Module	Comments
DSQ	An RTE-A module required to be relocated in the system generation in order to initialize NS-ARPA/1000. This module re-queues class I/O messages. The relocatable file is `/RTE_A/%DSQ`.
INPRO	(*Transport Monitor*) Inbound message processor for NS-ARPA/1000 transport. *Requires one class number*.
OUTPRO	(*Transport Monitor*) Outbound message processor for NS-ARPA/1000 transport. *Requires one resource number and one class number*.
MMINIT	Memory Manager initialization program. Sets up DSAM.
NSINIT NSINIT.MSG	NSINIT and its associated modules initialize and shut down the network. NSINIT.MSG is the NSINIT message file, and is optional. However, HP recommends that you install this message file. *NSINIT requires one resource number* while initializing or shutting down NS-ARPA.
UPLIN	UPLIN cleans up NetIPC resources. It is the timeout and re-enable module that maintains a running time on all transactions, artificially terminates (“times out”) any transaction that is not serviced within a user-specifiable time limit and cleans up Transaction Control Blocks (TCBs), NetIPC user records, sockets, and resource numbers. It can restart any HP-supplied slave monitor that has been aborted, and log off HP 3000 or HP 1000 sessions whose creating program has terminated with a session still outstanding. *HP recommends that you assign UPLIN to its own partition*.

Recommended for All Nodes

HP recommends that you provide the following services at all nodes:

Event Logging

Module	Comments
BREVL	Terminates event logging.
EVMON	Event monitor. Logs disasters, errors, warnings, and internal state information. *HP recommends that you RP EVMON before initializing NS-ARPA/1000*.
LOGCHG	Changes the event log mask, which determines which classes of events that EVMON will log.

Nodal Registry Utilities

Module	Comments
NRINIT	Nodal Registry initialization program. Used to specify communications with nodes that do not have all IEEE 802.3 LAN links or with non-HP nodes.
NRLIST	Nodal Registry List program. Lists the contents of the Nodal Registry.

Tracing

Module	Comments
NSTRC	NS-ARPA trace utility. Records messages at the socket and network level. If the NSTRC program is loaded with its working set size equivalent to its VMA size, the entire trace file will be in memory when NSTRC runs. If the NSTRC VMA size is larger than its working set size, there will be a page fault every 10 trace records, and NSTRC may drop trace records. *Requires two class numbers*.
BRTRC	Halts NS-ARPA tracing.
FMTRC	Formats trace files produced by NSTRC. Must be loaded with the same VMA size as NSTRC.

Information Utility

Module Comments

NSINF NS-ARPA Information Utility. Prints local address information, configured NS-ARPA resources, NS-ARPA/1000 network management utilities status and statistics, Memory Manager statistics, NS-ARPA program information, DS/1000-IV Compatible Services information, Nodal Routing Vector information, Remote Session information, and Message Accounting information.

PING Level 3 loopback. Performs a level 3 (IP) loopback and prints the results on the terminal screen. Includes each packet's round-trip time and summary of round-trip times.

Related Subsystem Modules

If you are using any of the following subsystems, you must relink the listed modules each time you install a new version of NS-ARPA/1000 to ensure compatibility. In addition, you must relink all user programs that access NS-ARPA or DS/1000-IV services.

Transparent File Access

Transparent File Access (TRFAS) is an RTE feature that allows you to interactively and programmatically access remote files (hierarchical and FMGR files).

Module Comments

TRFAS (*Slave Monitor*) Transparent File Access monitor. TRFAS is part of the RTE operating system and is required at the node at which the file resides. Load as a system utility (SU) and with access to system labeled common (LC). Search `NSSYS.LIB` and `NSLIB.LIB`.
Requires one class number.

DSRTR Transparent file access master. DSRTR is part of the RTE operating system and is required at the node at which the file access request originates. Load as a system utility (SU) and with access to system labeled common (LC). Search `NSSYS.LIB` and `NSLIB.LIB`.

Remote Image/1000 Access

Module Comments

RDBAM Remote Database Access monitor. Search `NSSYS.LIB` and `NSLIB.LIB`.
Requires one class number.

Required for Specific Links

The following modules are required for specific links, regardless of the services used. This list does not include link interface drivers; refer to "System Generation" for driver information.

IEEE 802.3 and/or Ethernet Links

Module	Comments
NRINIT (optional)	Nodal Registry initialization program. If the node belongs to more than one network, you must initialize the node in the Nodal Registry with NRINIT.

HP also recommends that you load the LAN/1000 Node Manager product. Refer to the *HP 12076A LAN/1000 Node Manager's Manual* for more information on using the LAN/1000 Node Manager product.

Router/1000 Links

Module	Comments
QUEUE	The NS-ARPA/1000 program scheduled by the interface driver to allocate a class buffer in SAM to receive incoming messages. *HP recommends that you assign QUEUE to its own partition*, especially if there is performance degradation.
QCLM	(*Transport Monitor*) Communications Error Logger. Prints most errors for DS/1000-IV Compatible Transport and Service Modules. *Requires one class number*.
GRPM	(*Transport Monitor*) RTE-RTE request/reply processor. The NS-ARPA/1000 program scheduled by ID*66 to allocate a class buffer in SAM to receive incoming messages. Routes incoming messages to Slave Monitors' class numbers. *Requires one class number*.
NRINIT	Nodal Registry initialization program. Not required for networks that only support DS/1000-IV Compatible Services over RTR links, Bisync, or X.25 (RTE-MPE) links.
DSMOD	(Recommended) Allows you to alter NS-ARPA parameters set by NSINIT during initialization. DSMOD allows the user to change the HP 3000 Bisync ID sequence, re-enable a link, display the Nodal Routing Vector (NRV), change the non-session password, schedule additional monitors, adjust timing, change the default session user name, and change the NRV.

X.25 Links

Refer to the X.25 manuals for complete information on the modules needed for X.25 links.

Required for Specific NS Features and Services

The following modules are required for specific NS Common Services features, excluding DS/1000-IV Compatible Services and ARPA Services.

Network File Transfer

NFT allows you to copy files to and from remote nodes.

At the Initiator node (the node that the request originates at), DSCOPY is required. DSCOPY establishes a VC connection with the monitor NFTMN at the Producer node (the node at which the file to be copied resides). NFTMN at the Producer node schedules PRODC. PRODC schedules PRDC1 if the user specified a file mask. PRODC establishes a VC connection with the monitor NFTMN at the Consumer node (the node to which the file will be copied). NFTMN at the Consumer node schedules CONSM.

Module	Comments
CONSM	Required at the Consumer node.
DSCOPY	Required at the Initiator (the node at which the NFT request is made). Establishes a VC connection with NFTMN at the Producer node.
NSERRS.MSG	NFT error message file. HP *recommends* that you install this file at all nodes.
NFTMN	NFT monitor program. NSINIT schedules NFTMN at initialization time and NFTMN remains scheduled, waiting for an `IPCRecvCn` call. At the Producer node, NFTMN schedules PRODC; at the Consumer node, NFTMN schedules CONSM. At the Initiator node, NFTMN is not required (unless the node also happens to be a Producer or Consumer node). However, a warning will occur when NSINIT starts up and schedules NFTMN, because NFTMN will not be at the Initiator node. *Requires one class number*.
PRODC	Required at the Producer node.
PRDC1	Required at the Producer node if file masks or directory extensions are used. Expands file masks and produces directory lists.

Remote Process Management (RPM)

Remote Process Management (RPM) enables a program on one NS-ARPA/1000 node to schedule a program on the local node or at a remote node. The execution of the scheduled program (known as the child program) can then be controlled or terminated through RPM.

Module	Comment
RPMMN	RPM monitor required at the node receiving RPM requests from parent programs to schedule or terminate child programs. *HP recommends that you RP RPMMN before initializing NS-ARPA/1000*.

Required for Specific ARPA Services

The following modules are required for specific ARPA Services.

INETD

The Internet Network Services Daemon (INETD) is required by the TELNET and FTP services.

Module	Comment
INETD	Internet Network Services Daemon that monitors all incoming connection requests for TELNET, FTP, SMTP (Mail/1000), Remote Print Spooling, and Time Server. Once a connection is established, INETD schedules the appropriate server program, TNSRV, FTPSV, SMTP, or RLP Daemon, and passes the connection to the server. INETD is required at the TELNET or FTP server node. See the “INETD—Internet Network Services Daemon” section for more information.

TELNET Virtual Terminal

TELNET provides a virtual terminal service between NS-ARPA/1000 nodes. It allows you to log on to a remote system.

Module	Comment
TELNET	Interactive user interface program. The TELNET user program runs on the client (local) node and communicates with the remote TELNET server.
TNSRV	TELNET server to provide remote access. The TELNET server program runs on the remote node and handles the virtual terminal session. Required at the TELNET server node.
ALARM SIGNAL	RTE-A modules required in the system generation for TELNET. The relocatable files are `/RTE_A/%ALARM` and `/RTE_A/%SIGNL`, respectively.

FTP File Transfer

FTP provides file transfer and file management operations between NS-ARPA/1000 and other ARPA nodes.

Module — Comment

FTP — Interactive user interface program. The FTP user program runs on the client (local) node and communicates with the remote FTP server.

FTPSV — FTP server to provide remote access. The FTP server program runs on the remote node and handles the FTP session. Required at the FTP server node.

Time Server Protocol

RDATE — Client program that reads time from remote system and sets system time in local system.

Required for HP 1000 File Server

The following module is required for the HP 1000 file server.

FSRV — The HP 1000 file server that allows remote NFS* client systems to mount and access the RTE-A file system.

Required for DS/1000-IV Compatible Services (RTE-RTE)

The following modules are required for DS/1000-IV (RTE-RTE) Compatible Services. Additional modules are required for specific features and services.

Module — Comments

QCLM — (*Transport Monitor*) Communications Error Logger. Prints most errors for DS/1000-IV Compatible Transport and Service Modules.
Requires one class number.

IFPM — (*Transport Monitor*) Interface protocol monitor. Interface between DS/1000-IV Compatible Services and LAN links for outbound messages. Required in all systems that will use DS/1000-IV Compatible Services over non-Router/1000 links.
Requires two class numbers. One class number is used by INPRO to move inbound messages from INPRO (DSAM) to IFPM (SAM).

GRPM — (*Transport Monitor*) RTE-RTE request/reply processor. Routes incoming messages to Slave Monitors' class numbers.
Requires one class number.

* NFS is a trademark of Sun Microsystems, Inc.

Required for Specific DS/1000-IV Compatible Services (RTE-RTE)

This subsection lists the modules required for specific DS/1000-IV Compatible Services (RTE-RTE), including slave monitors required when the local node is the destination node for NS-ARPA/1000 subroutine calls. You can exclude software modules if your node does not need the services it provides, but note that remote master programs might require slave monitors such as OPERM and EXECM at the destination HP 1000. If a request is made to a node that does not have the required slave monitor, the request will be returned with a `DS06` error, illegal request.

Network Parameter Modification

Module	Comments
DSMOD	Allows you to alter NS-ARPA parameters set by NSINIT during initialization. DSMOD allows the user to change the HP 3000 Bisync ID sequence, re-enable a link, display the Nodal Routing Vector (NRV), schedule additional monitors, adjust timing, change the default session user name, and change the NRV. *HP recommends that you install DSMOD*.

Message Accounting (MA)

Message Accounting is a datagram-oriented protocol that provides message reliability by retransmitting lost messages and suppressing duplicate messages. *HP recommends that you use Message Accounting*.

Module	Comments
MATIC	Provides time-out processing for Message Accounting. *Requires one resource number*.

Dynamic Rerouting (Router/1000 Links only)

Dynamic Rerouting automatically reroutes messages around inoperative links without user intervention. Dynamic Rerouting is an option provided for HDLC links configured as Router/1000 links only. HP recommends that you use Dynamic Rerouting with these links.

Module	Comments
#SEND	Used with Dynamic Message Rerouting. Sends update messages to neighboring nodes.

Distributed EXEC (DEXEC)

There are two slave monitors to service DEXEC requests from remote nodes.

Module	Comments
EXECM	(*Slave Monitor*) Remote EXEC monitor. Services remote EXEC (DEXEC) calls. This module must be present in all RTE destination nodes where any remote EXEC calls will be processed. These requests may come from other HP 1000 or HP 3000 nodes. *Requires one class number.*
EXECW	(*Slave Monitor*) Remote "schedule with wait" monitor (with or without queuing). Services remote EXEC (DEXEC) requests to schedule programs with wait. To run a program from REMAT or to execute the `LO` (LOAD) and `PL` (PROGRAM LIST) commands on memory based nodes, the remote node must have EXECW. *Requires one class number.*

The following subroutine calls require either EXECM or EXECW at the destination node:

Subroutine	Required Slave Monitor
DEXEC (1)	EXECM
DEXEC (2)	EXECM
DEXEC (3)	EXECM
DEXEC (6)	EXECM
DEXEC (9)	EXECW
DEXEC (10)	EXECM
DEXEC (11)	EXECM
DEXEC (12)	EXECM
DEXEC (13)	EXECM
DEXEC (23)	EXECW
DEXEC (24)	EXECW
DEXEC (25)	EXECM
DEXEC (99)	EXECM

Program-to-Program Communication

The following calls are Program-to-Program (PTOP) master calls:

```
    POPEN   PWRIT   PREAD   PCONT   PCLOS   PNRPY
```

The following are PTOP slave calls:

```
    GET   ACEPT   REJCT   FINIS
```

If the slave-side node is an RTE, PTOPM is required in the slave node. If the slave-side node is an HP 3000, NS/3000, or DS/3000, software must be installed at the slave node.

Module	Comments
PTOPM	(*Slave Monitor*) PTOP Communication slave monitor. Handles programmatic POPEN, PREAD, PWRITE, PCONT, and FINIS requests and REMAT commands `SO` (SLAVE OFF) and `SL` (SLAVE LIST) on the slave side. *Requires one class number.*

REMAT

REMAT provides remote and local FMGR file manipulation capability, and sends RTE commands to any remote RTE or the local system. It can be used to provide almost unattended operation of other nodes (system error messages will still be printed at the local operator's console unless the system console is mapped to another node via Remote I/O Mapping).

Module — Comments

REMAT — The REMAT command interface.

In addition, the following slave monitors are required for the local node to respond to requests from remote nodes. Refer to Table 4-3 to determine the specific slave monitors required for REMAT commands.

Module — Comments

APLDR — Absolute program loader, required at the target or destination node which is a memory-based system. Processes REMAT `IO`, `LO`, `PL` commands. At the source node, RFAM transfers the linked program to the destination node. At the destination node, EXECW schedules APLDR to load programs at the target node.

APLDR also processes FLOAD utility calls. If you generate in %LOAD or %MEMRY, you can not use the REMAT `LO` command or the FLOAD utility.

DLIST — (*Slave Monitor*) Remote Directory List. Lists contents of FMGR directories.
Requires one class number.

EXECM — (*Slave Monitor*) Remote EXEC monitor. Services remote EXEC (DEXEC) calls. This module must be present in all RTE destination nodes where any remote EXEC calls will be processed.
Requires one class number.

EXECW — (*Slave Monitor*) Remote "schedule with wait" monitor (with or without queuing). Services remote EXEC (DEXEC) requests to schedule programs with wait. This program must be present in all RTE nodes to run a program from REMAT or to execute the `LO` (LOAD) and `PL` (PROGRAM LIST) commands on memory based nodes.
Requires one class number.

OPERM — (*Slave Monitor*) Remote RTE operator command capability.
Requires one class number.

RFAM — (*Slave Monitor*) Remote File Access monitor (for FMGR files only). Required at the node at which the file resides. The relocatable file is %RFAM2.
Requires one class number.

Table 4-3. REMAT commands and Required Slave Modules

Command	Slave Monitor Required at Node 1	Slave Monitor Required at Node 2
AT		RSM (if SW to an account)
BC	EXECM (at all receiving nodes)	EXECM (at all receiving nodes)
CL	DLIST	
CR	RFAM, EXECM	
DE		RSM (if SW to an account)
DL	DLIST	
DU	RFAM	EXECM
FL	RFAM	
IO	APLDR requires EXECM at operator's node to print results and error messages EXECW	
LI	RFAM	EXECM
LO	RFAM requires EXECM at operator's node to print results and error messages	APLDR, EXECW
PL	APLDR requires EXECM at operator's node to print results and error messages	
PU	RFAM	
QU	EXECM	
QW	EXECW	
RN	RFAM	
RW	EXECW	
SD	RSM	
SL	PTOPM	
SO	PTOPM	
ST	RFAM, EXECM	RFAM
SW		RSM (if SW to an account)
TE	EXECM	
TR	TRFAS is required at the node where the transfer file exists if remote file. DSRTR is required on the local node.	

Note You should include OPERM at all nodes that support DS/1000-IV Compatible Services. RSM is a Remote Session monitor used only in Session Monitor nodes with DS/1000-IV.

Remote Database Access

Remote Database Access (RDBA) allows you to access IMAGE/1000 databases from remote nodes. For more information, refer to the appropriate IMAGE reference manual.

Module	Comments
RDBAM	(*Slave Monitor*) Remote Database Access monitor. Search `NSSYS.LIB` and `NSLIB.LIB`. *Requires one class number.*

Remote File Access (FMGR Files)

Remote File Access (RFA) allows you to programmatically access remote FMGR files. For remote access of hierarchical files, refer to the subsection "Transparent File Access."

Module	Comments

RFAM	(*Slave Monitor*) Remote File Access monitor. RFAM is required when the following RFA subroutines are used for a file that exists at the local node:
	`DCRET DPURG DOPEN DCLOS DREAD DWRIT`
	`DPOSN DWIND DNAME DCONT DLOCF DAPOS DSTAT`
	`DXPOS DXLOC DXAPO DXCRE DXCLO DXREA DXWRI`
	The relocatable file is %RFAM2. %RFAM2 can have several local files simultaneously open from remote nodes. For each file that it has open, RFAM must allocate a Data Control Block (DCB), which is 153 words long. RFAM allocates DCBs from its internal buffer and from the remainder of its program partition. RFAM should be assigned to a reserved partition so that the number of DCBs will be a constant amount. You can size up RFAM's program partition to accommodate more DCBs.
	Requires one class number.

Remote I/O Mapping

Remote I/O Mapping redirects I/O requests to remote HP 1000 nodes.

Module	Comments
IOMAP	User interface for Remote I/O Mapping.
LUMAP	DEXEC Request Module for Remote I/O Mapping. *Requires one class number.*
LUQUE	Provides class buffers for Remote I/O mapped data transfer.
SYSAT	System Attention Module required for Remote I/O Mapping. Sends a message to a remote system to set a program's break flag. Also used to send the System Attention request to a remote system. SYSAT requires OPERM to be scheduled or available at the remote node.

Remote Program Download

Remote Program downloads allow you to transfer programs (type 6 files) to remote systems. The program files must be FMGR files. With Remote Program Downloads you can load programs to remote memory-based systems, without storing source files at the memory-based system.

Module	Comments
APLDR	Processes REMAT `IO`, `LO`, and `PL` commands.
EXECW	(*Slave Monitor*) Required at the destination node. Remote “schedule with wait” Monitor. Schedules APLDR. *Requires one class number.*
RFAM	(*Slave Monitor*) Required at the source node. Remote File Access monitor. Transfers linked program to destination node. *Requires one class number.*

Remote System Downloads

Remote System Downloads allow you to load systems to remote DS/1000-IV nodes for nodes without operators or terminal-less nodes in harsh environments. Forced Cold Loads allow you to halt the system and download a new operating system.

The node that you are downloading a DS/1000-IV system to must be connected by a Router/1000 HDLC.

Module	Comments
DSVCP	DS Virtual Control Panel operator interface module for remote control of the A-Series front panel.
PROGL	(*Slave Monitor*) Slave monitor for remote system download. Can simultaneously handle requests from up to 20 nodes at the same time. User supplied subroutines enable store and forward and/or LU-to-file-conversion capabilities. Use `%PROGZ` for store and forward capability; use `%PROGL` otherwise. *Requires one class number.*
VCPMN	(*Slave Monitor*) Virtual Control Panel Monitor. This module monitors the Virtual Control Panel of a remote A-Series CPU. *Requires one class number.*

Remote Virtual Control Panel

Module Comments

DSVCP DS Virtual Control Panel operator interface module for remote control of the A-Series front panel.

VCPMN (*Slave Monitor*) Virtual Control Panel Monitor. This module monitors the Virtual Control Panel of a remote A-Series CPU.
Requires one class number.

Utilities

The following utilities require the slave monitors listed at the destination node:

Utility	Slave Monitor Required	
DMESG	EXECM	
DMESS	OPERM	
DSERR	none	
FCOPY	RFAM	At both the origination and destination node
FLOAD	APLDR	Force-loads a program (need RFAM) into a remote memory-based node (need APLDR and EXECW)
	EXECW	
	RFAM	Located at the node where program file(s) reside
SEGLD	RFAM	Located at the node where program file(s) reside

Required for DS/1000-IV Compatible Services (RTE-MPE)

The following modules are required for any RTE-MPE communication over DS/1000-IV Compatible Links (Bisync or X.25). Additional modules are required for specific services.

Module Comments

QUEUE The NS-ARPA/1000 program scheduled by the interface driver to allocate a class buffer in SAM to receive incoming messages. Queues incoming messages on the appropriate slave monitor's class number.
HP recommends that you assign QUEUE to its own partition.

QUEX (*Transport Monitor*) HP 3000 communication monitor.
Requires one class number.

QUEZ (*Transport Monitor*) HP 3000 I/O completion monitor (PSI).
Requires one class number.

QCLM (*Transport Monitor*) Communications Error Logger. Prints most errors for DS/1000-IV Compatible Transport and Service Modules.
Requires one class number.

Bisync Links

In addition, the following module is required for Bisync links:

Module	Comments
DSLIN	Establishes PSI BISYNC connection to 3000s.

Slave Services

In addition, the following modules are required if the HP 1000 is to be the slave node in any transactions (PTOP slave program, recipient of RFA, DEXEC, or remote operator requests).

Module	Comments
RPCNV	(*Converter*) NS-ARPA/1000 to DS/3000 reply converter. *Requires one class number*.
RQCNV	(*Converter*) NS-ARPA/1000 to DS/3000 request converter. *Requires one class number*.

Recommended for DS/1000-IV (RTE-MPE) Compatible Services

HP recommends that you load the following modules in all HP 1000 nodes with RTE-MPE communication over DS/1000-IV Compatible Links (Bisync and X.25).

Module	Comments
LOG3K	Provides operator control over recording of DS messages to and from HP 3000s. You must use a magnetic tape unit as a log device.
TRC3K	Formats the data recorded by LOG3K.
CNSLM	(*Slave Monitor*) HP 3000 $STDLIST Monitor. Reports MPE TELL and WARN messages. *Requires one class number*.

Specific DS/1000-IV Compatible Services (RTE-MPE)

The following modules are required or recommended for specific DS/1000-IV Compatible Services (RTE-MPE). These services are supported only over DS/1000-IV Compatible Links (Bisync or X.25).

Operator Commands

The following module is required to process RTE operator commands entered from the HP 3000.

Module	Comments
OPERM	(*Slave Monitor*) Handles commands from the HP 3000. *Requires one class number*.

RMOTE

RMOTE provides “virtual terminal” capability, which allows an RTE terminal to appear as though it were connected directly to an HP 3000.

Module	Comments
RMOTE	The relocatable file is `%RMOT1`.
MVCP3	Used to install the PTOP slave program `COPY3K.PUB.SYS` on an HP 3000 for use in implementing the RMOTE `MO` (MOVE) command. The file `!COPY3` is required by MVCP3.

MPE TELL and WARN Messages

Module	Comments
CNSLM	(*Slave Monitor*) HP 3000 $STDLIST Monitor. Reports MPE TELL and WARN messages. *Requires one class number*.

Distributed EXEC (DEXEC)

Module Comments

EXECM (*Slave Monitor*) Remote EXEC Monitor. Services remote EXEC (DEXEC) calls.
Requires one class number.

EXECW (*Slave Monitor*) Remote “schedule with wait” Monitor (with or without queuing). Services remote EXEC (DEXEC) requests to schedule programs with wait.
Requires one class number.

Program-to-Program Communication (HP 1000 Slave)

Module Comments

PTOPM (*Slave Monitor*) PTOP Communication slave monitor. Handles programmatic POPEN, PREAD, PWRITE, PCLOS, PCONT, and FINIS requests.
Requires one class number.

DSTES PTOP slave program (master DSTEST on HP 3000). Verifies the PTOP software.

Remote File Access (FMGR Files)

Module Comments

RFAM (*Slave Monitor*) Remote File Access monitor. Handles RFA requests from HP 3000. Required at the node at which the file resides.
Requires one class number.

NS-ARPA Software Summary and Guidelines

NS-ARPA software modules are listed below, according to NS-ARPA activities. Unless otherwise specified, all the modules are required for the given activity.

When you run NSINIT, the modules marked [1] must be in one of the following situations:

- in the `/PROGRAMS` directory
- in the current working directory
- in a directory in the User-Definable Search Path (UDSP #1)
- restored (via the RTE `RP` command)

The modules marked [2] must be restored. Otherwise, NSINIT aborts NS-ARPA initialization.

The modules marked [3] must be restored before NSINIT attempts to schedule them for NS-ARPA initialization if you want to use those modules.

The modules marked [4] are message files that must be on the directory `/SYSTEM` when you run NSINIT if you want to use them.

The modules marked [5] are programs that are cloned for each connection; do *not* restore (`RP`) them. Install them in `/PROGRAMS`.

The modules marked [6] must be restored in the system session before they are used. NSINIT does not check if they have been restored.

NS-ARPA/1000 supplies a command file that links all NS-ARPA/1000 programs and places them in the appropriate directories. The file is `/NS1000/CMD/INSTALL_NS1000.CMD` ; for more information, refer to Section 6, "NS-ARPA/1000 Program and File Installation."

All Nodes

BREVL
BRTRC
EVMON[3]
FMTRC
INPRO[2]
LOGCHG
MMINIT[1]
NRINIT
NRLIST
NSINF
NSINIT
NSINIT.MSG[4]
NSTRC[3]
OUTPRO[2]
UPLIN[2]

Transparent File Access (DS Transparency)

DSRTR[6]	(part of RTE)
TRFAS[3]	(part of RTE)

Remote Image/1000

RDBAM[3]	(part of Image/1000)

Router/1000 Links

QUEUE[2]
QCLM[3]
GRPM[2]
DSMOD

X.25 Links

XNET (part of the X.25/1000 product)

Network File Transfer

CONSM[5]
DSCOPY
NFTMN[3]
PRODC[5]
PRDC1[5]
NSERRS.MSG[4]
DSCOPY.HLP[4]

Remote Process Management (RPM)

RPMMN[3]

TELNET

TELNET
TNSRV[5]
INETD[3]

File Server

FSRV

FTP

FTP
FTPSV[5]
INETD[3]

Time Server Protocol

RDATE

DS/1000-IV Compatible Services (RTE-RTE)

QCLM[3]

IFPM[3] (non-Router/1000 links only)

GRPM[2]

DSMOD

Message Accounting (recommended)

MATIC[2]

Dynamic Rerouting

#SEND[2]

Distributed EXEC

EXECM[3]

EXECW[3]

Program-to-Program Communication

PTOPM[3]

REMAT

REMAT[3]

APLDR[3]

DLIST[3]

EXECM[3]

EXECW[3]

OPERM[3]

RFAM[3]

Remote Database Access

RDBAM[3]

Remote File Access (FMGR Files)

RFAM[3]

Remote I/O Mapping

IOMAP[6]
LUMAP[6]
LUQUE[6]
SYSAT

Remote Program Download

APLDR
EXECW[3]
RFAM[3]

Remote System Download

DSVCP
PROGL[3] (%PROGL for regular operation, %PROGZ for store and forward)
VCPMN[3]

Remote Virtual Control Panel

DSVCP
VCPMN[3]

Utilities

APLDR[3]	(for FLOAD and SEGLD)
EXECM[3]	(for DNODE and DMESG)
EXECW[3]	(for FLOAD)
OPERM[3]	(for DMESS)
RFAM[3]	(for FCOPY)

DS/1000-IV Compatible Services (RTE-MPE)

QUEUE[2]
QUEX[2]
QUEZ[2]
QCLM[3]
LOG3K
TRC3K

Bisync Links

```
    DSLIN[3]
```

All Slave Services

```
    RPCNV[3]
    RQCNV[3]
```

Remote Operator

```
    OPERM[3]
```

RMOTE

```
    RMOTE    (%RMOT1)
    MVCP3
```

MPE TELL and WARN Messages

```
    CNSLM[3]
```

Distributed EXEC

```
    EXECM[3]
    EXECW[3]
```

Program-to-Program Communication

```
    PTOPM[3]
    DSTES[6]
```

Remote File Access (FMGR Files)

```
    RFAM[3]
```

System Resources

This subsection describes the system resources required for NS-ARPA/1000.

ID Segments

There are about 70 NS-ARPA subsystem programs. Each program requires an ID segment.

60 to 80 ID segments are needed for all NS-ARPA/1000 services and connections excluding DS/1000-IV Compatible Services (RTE-MPE). This amount includes the three ID segments required for the NS-ARPA initialization programs (NSINIT, MMINIT, and NRINIT) and ID segments required to run network manager utilities. Because the NS-ARPA initialization programs and network manager utilities are not always active, you may not have to allocate additional ID segments for these programs. Instead, it is possible to use ID segments from a pool shared by other programs.

Each FTP server or user program uses one ID segment.

Each TELNET server or user program uses one ID segment.

Class Numbers

A sufficient number of class numbers is 30. The minimum is four class numbers if DS/1000-IV Compatible Services are not generated into NS-ARPA. Otherwise, the minimum is seven to support DS/1000-IV Compatible Services. The following software modules each require one class number:

```
CNSLM    EXECW    NFTMN    QCLM     RPCNV
DLIST    GRPM     OPERM    QUEX     RQCNV
EXECM    LUMAP    PROGL    QUEZ     TRFAS
                  PTOPM    RDBAM    VCPMN
```

IFPM requires two class numbers. One class number is used by INPRO to move inbound messages from INPRO (DSAM) to IFPM (SAM).

NSTRC requires one class number for network-level tracing and an additional class number for socket-level tracing.

When you run NSINIT and initialize a node, NSINIT prints the number of class numbers required for NS-ARPA monitors according to the services configured. This number includes class numbers for all monitors specified for scheduling, whether or not the monitor is present and restored (RPed).

In addition, the following conditions require class numbers:

- NSINIT uses 5 class numbers while it initializes or shuts down NS-ARPA, and releases them when it terminates.
- Each slave PTOP program requires one class number, which is allocated automatically in the slave program's node when the program is opened and returned when the slave terminates, via either `FINIS` or `PCLOS`.

- Master programs (programs that call any RFA, PTOP routines, or DEXEC, FLOAD, DMESS, DMESG, FCOPY, and so forth) require a class number for the duration of the request. These class numbers are automatically allocated by the NS-ARPA/1000 master routines, and are returned when the request completes. Thus, these class numbers can be shared among programs.
- Each active REMAT and RMOTE clone requires one class number.
- For RTE-MPE communication, each message that requires a continuation buffer requires one class number.
- The HP 12076A LAN/1000 Node Manager product requires two class numbers. (HP recommends that you use this product with NS-ARPA/1000 if you have LAN links.)
- Each TELNET user interface program requires one class number for I/O purposes.

FTP does not require any class numbers.

Resource Numbers

A sufficient number of resource numbers is 20.

The following programs require one system resource number each:

INETD
EVMON
OUTPRO
DS/1000-IV Compatible Services
Message Accounting
Every FTP and TELNET user and server program
FSRV

Each active NetIPC process requires one system resource number.

5

System Generation

This section describes the NS-ARPA/1000 answers needed for an RTE-A system generation. An example system generation answer file is provided in file `/NS1000/EXAMPLES/#ANSNS` .

Note You must keep an up-to-date copy of the generation answer file, either on paper or disk. Should you have any problems with your system, HP will require the generation answer file.

System Generation Preparation

Your RTE-A system should be a good working system before you regenerate to create an NS-ARPA/1000 system. Follow these precautionary steps before regenerating the system:

1. Back up your current system, and verify your backup copy.
2. Look at the example answer file, `/NS1000/EXAMPLES/#ANSNS` . Determine how you want to regenerate your system.
3. Determine the internal resources required for NS-ARPA (Refer to Section 4, “Internal Resources”). This includes memory, system resources, and software modules.
4. Determine the number of TELNET pseudo terminal LUs and modem LUs that will be controlled by HPMDM.

   If more than eight is desired, you must reconfigure the `HPMDM_TABLE` table to increase the number of LUs before system generation. Refer to “TELNET Pseudo Terminal LU Initialization” of Section 12, “Additional Initialization Tasks” for more information. Refer to that subsection first before proceeding with your system generation.

   During system generation, `HPMDM_TABLE.REL` is to be relocated into labeled system common. Refer to “Labeled Common” later in this section.
5. Modify your RTE-A system generation answer file for your link(s) and NS-ARPA/1000 services as described in this section.

System Generation Overview

The remainder of this section provides detailed information on the system generation overview steps listed below:

1. System Relocation Phase.

   ```
   RE,/NS1000/REL/NSABP.REL
   RE,/RTE_A/%DSQ
   RE,/RTE_A/%ALARM
   RE,/RTE_A/%SIGNL
   RE,/RTE_A/XMB.REL,MB02    For A900 pre-Rev.4 firmware only
   RE,/RTE_A/XMB.REL,MB12    For A900 pre-Rev.4 firmware only
   ```

   If you also have an X.25 link, relocate:

   ```
   RE,/NS1000/DSREL/%CXL66
   RE,/X25/REL/CSTB.REL
   RE,/RTE_A/XMB.REL,MB01    For A900 pre-Rev.4 firmware only
   RE,/RTE_A/XMB.REL,MB10    For A900 pre-Rev.4 firmware only
   ```

2. Driver Relocation or Driver Partition Phase.

   ```
   RE,/NS1000/DSREL/%ADV00
   RE,/RTE_A/%ID*67
   RE,/NS1000/DSREL/%ID*66
   RE,/RTE_A/IDZ00.REL
   RE,/RTE_A/DDC00.REL
   ```

3. Table Generation Phase.

   For each LAN card, enter one IFT entry and one DVT entry.

   For each HDLC or Bisync card, enter one IFT entry and two DVT entries.

   Up to three LAN cards are allowed for your NS-ARPA/1000 node.

   For TELNET, enter one IFT entry overall. For each TELNET pseudo terminal LU, enter one DVT entry.

   For Remote I/O Mapping, enter one IFT entry and one reserved DVT entry. For each mappable LU, enter one DVT entry.

   For each X.25 card, enter one IFT entry and two DVT entries. For the DDX00 driver, enter one reserved IFT entry. For XNET, enter two reserved DVT entries. For each switched or permanent X.25 VC (Virtual Circuit/Connection), enter two DVT entries.

4. Memory Allocation.

   Check the system resources, such as resource numbers, class numbers, SMB, to make sure they are adequate. Relocate the following NS-ARPA/1000 modules into labeled system common:

   ```
   RE,/NS1000/DSREL/%RESA
   RE,/RTE_A/HPMDM_TABLE.REL
   ```

You must search the following NS-ARPA/1000 library to resolve external references in %RESA as follows:

```
    SE,/NS1000/LIB/NSSYS.LIB
```

After the system generation, you can boot your new system and install NS-ARPA/1000 programs. Refer to the next sections, "NS-ARPA/1000 Program and File Installation" and "BOOT and WELCOME Files."

Create the /SCRATCH directory. FTP uses the /SCRATCH directory for temporary files. The directory must not be write-protected.

```
    CI> CRDIR /SCRATCH lu
```

You can create the /SCRATCH directory before or after you boot your new system.

System Relocation Phase

You must relocate the following modules in the system area:

Module	Comment
/NS1000/REL/NSABP.REL	NS-ARPA abort processor. NSABP is partitionable.
/RTE_A/%ALARM	RTE-A alarm module for Signals/1000. %ALARM is part of the RTE operating system and is partitionable.
/RTE_A/%DSQ	Requeues class I/O messages. %DSQ is part of the RTE operating system and is partitionable.
/RTE_A/%SIGNL	RTE-A Signals/1000 module. %SIGNL is part of the RTE operating system and is partitionable.
/RTE_A/XMB.REL,MB02 /RTE_A/XMB.REL,MB12	These modules are needed only on A900 systems with firmware older than Rev. 4. They contain entry points for the cross map move byte instructions which are NOT in the A900 RPL file, %RPL91. They are referenced by the TELNET pseudo terminal driver, IDZ00, and the X.25 pseudo-device driver, DDX00. If these modules are not relocated, an undefined external error is displayed.

X.25 Links

For X.25 links, you must relocate the following modules:

Module	Comment
/NS1000/DSREL/%CXL66	NS-ARPA/1000 customizing subroutine. Alternately, you can place %CXL66 in a driver partition with the X.25 pseudo-device driver, DDX00.REL (if partitioned, they must be in the same driver partition).
/X25/REL/CSTB.REL	X.25 customizing subroutine table for NS-ARPA. (Part of the X.25 product.)
/RTE_A/XMB.REL,MB01 /RTE_A/XMB.REL,MB10	These modules are needed only on A900 systems with firmware older than Rev. 4. They contain entry points for the cross map move byte instructions which are NOT in the A900 RPL file, %RPL91. They are referenced by the X.25 pseudo-device driver, DDX00.

Partitioned Command

If NSABP will be relocated in an OS module partition, you must use the PA command to inhibit the generator from using the dummy module NSA.. from $SYSA.

```
    PA, NSABP
```

Driver Relocation Phase

You can relocate NS-ARPA/1000 drivers in the system. However, this is not necessary since all NS-ARPA drivers are partitionable. The NS-ARPA drivers are discussed in the following subsection, and listed in Table 5-1.

Driver Partition Phase

Because all NS-ARPA/1000 drivers are partitionable, you can relocate them during this phase. Relocate drivers for the links and services you need. Table 5-1 summarizes the drivers required for NS-ARPA links, TELNET, and Remote I/O Mapping and gives approximate page sizes for the drivers.

Table 5-1. Driver Partition Summary

Link/Feature	Relocatable Driver	Approx. # of pages
HDLC or BISYNC	`/NS1000/DSREL/%ID*66`	1.5
IEEE 802.3 or Ethernet LAN	`/RTE_A/%ID*67`	2
X.25	`/RTE_A/%IDS00`	1.25
	`/X25/REL/DDX00.REL`	2.5
	`/X25/REL/DD*60.REL`	0.75
	`/NS1000/DSREL/%CXL66`	1
Remote I/O Mapping	`/NS1000/DSREL/%ADV00`	1
TELNET pseudo terminal driver	`/RTE_A/IDZ00.REL`	1

RTE-A serial device driver needed by TELNET	`/RTE_A/DDC00.REL`
	`OR /RTE_A/DDC01.REL`
		1

`DDC01.REL` can be used instead of `DDC00.REL`. While `DDC01.REL` was designed to control slave devices—such as a printer—attached to a terminal, the use of slave devices over a TELNET connection is not supported.

Table Generation Phase

The table generation phase of the system generation sets up interface and device tables for the system. The interface table (IFT) entries identify the interface cards in the system and include such information as the interface driver and the card's select code. The IFT handles all interface I/O requests. The device table (DVT) entries contain device specific information, and establish LU numbers for devices.

LU Assignments for DS/1000-IV Compatible Services

For DS/1000-IV Compatible Services, you can only use LUs less than or equal to 63, with the following two exceptions:

1. Communication links can have LUs greater than 63.

2. Because programs can address their scheduling terminal as LU 1, terminals from which DS/1000-IV Compatible Services are scheduled can have LUs greater than 63.

For example, you can run REMAT from a terminal that is LU 78 and store data entered at the terminal to a file (by specifying LU 1 as the input LU). You could also store the data to any other device with an LU less than 63. However, you could not run REMAT and store data from your terminal to a printer that was LU 78.

DS/1000-IV Compatible Services Logging

DS/1000-IV Compatible Services write several logging messages to system LU 1, so you should associate this LU with an *unbuffered* device if you support DS/1000-IV Compatible Services. Do not associate the system LU 1 to a buffered device; if the system runs out of SAM, NS-ARPA cannot post messages to a buffered device.

Logging and Tracing Disk LUs

If you partition your disks, be sure to allocate disk LUs with adequate space for any networking log or trace files you may use.

LAN Links

You can use up to three LAN link interface cards for NS-ARPA/1000 per system. Each LAN link requires one IFT entry and one DVT entry.

Format

```
    IFT,/RTE_A/%ID*67,SC: selectCode
    *
    DVT,,,LU:lu,DT:67B,TO:200
```

Parameters

selectCode The select code of the LAN interface card.

lu The logical unit number which will be associated with the LAN interface card.

The IFT default parameter is contained in the %ID*67 file. For more information, refer to the *HP 12076A LAN/1000 Link LANIC Installation Manual*.

Example

```
* IEEE 802.3 LOCAL AREA NETWORK, LU 96
*
IFT,/RTE_A/%ID*67,SC:37B
*     dvt table entries for id*67 (12076A 802.3 LAN card)
*
DVT,,,LU:96,DT:67B,TO:200
```

HDLC and Bisync Links

Each HDLC or Bisync link requires one IFT entry and two DVT entries. The first DVT entry is for the *transmit LU*, which is the logical communication channel for transmitting messages. The second DVT entry is for the *receive LU*, which is the logical communication channel for receiving messages. The interface driver uses the DVTs to monitor activity on the communication link, which is full duplex. The transmit LU defined in each pair of DVTs is also used by NS-ARPA/1000 network administration programs (such as NSINIT and NSINF).

Format

```
    IFT,/NS1000/DSREL/%ID*66,SC: selectCode
    *
    DVT,,,LU:transmitLU,DT:66B
    DVT,,,LU:receiveLU,DT:66B
```

Parameters

selectCode — The select code of the HDLC or Bisync interface card.

transmitLU — The transmit LU for the communication channel.

receiveLU — The receive LU for the communication channel. Must equal *transmitLU* + 1.

Do not specify the number of extent words; use the default value in %ID*66 instead.

Example

```
* NETWORK LINKS  SELECT CODE = 24B
*                LU 79-80
*
IFT,/NS1000/DSREL/%ID*66,SC:24B
*
DVT,,,LU:79,DT:66B
DVT,,,LU:80,DT:66B
```

TELNET

The TELNET pseudo terminal driver, `IDZ00.REL`, requires one IFT entry overall and one DVT entry for each pseudo terminal LU. Each pseudo terminal LU corresponds to a virtual terminal connection for a remote node.

Format

```
*TELNET pseudo terminal driver
IFT,/RTE_A/IDZ00.REL
*TELNET pseudoterminal LUs
DVT,/RTE_A/DDC00.REL,MHP_TELNET,LU: telnetLU
DVT,...
  .
  .
  .
DVT,...
```

Parameters

telnetLU The pseudo terminal LU for each TELNET user allowed on the system.

For more information about the DVT parameters, refer to the serial driver in the *RTE-A Driver Reference Manual*, part number 92077-90011.

The version of HPMDM_TABLE.REL supplied with RTE-A contains entries for up to eight LUs (pseudo terminal LUs and modem LUs). TELNET uses pseudo terminal LUs. If you need to increase the number of LUs, refer to “TELNET Pseudo Terminal LU Initialization” of Section 12, “Additional Initialization Tasks” for more information. The TELNET pseudo terminal LUs must be initialized every time you reboot. During system generation, HPMDM_TABLE.REL is to be relocated into labeled system common.

The maximum number of TELNET server programs is 64 even if more pseudo terminal LUs are defined here. If there are no pseudo terminal LUs generated into the system, then the number of TELNET server programs is four. The number of TELNET *user* programs is determined by the number of terminals (not pseudo terminals) generated into the system with a maximum of 24 or a default of four.

Driver `DDC01.REL` can be used instead of `DDC00.REL`. While `DDC01.REL` was designed to control slave devices—such as a printer—attached to a terminal, the use of slave devices over a TELNET connection is not supported.

Example

```
*
*  4 TELNET LUs : 140 - 143
*
IFT,/RTE_A/IDZ00.REL
DVT,/RTE_A/DDC01.REL,MHP_TELNET,LU:140
DVT,/RTE_A/DDC01.REL,MHP_TELNET,LU:141
DVT,/RTE_A/DDC01.REL,MHP_TELNET,LU:142
DVT,/RTE_A/DDC01.REL,MHP_TELNET,LU:143
*
```

Remote I/O Mapping

Remote I/O Mapping requires one IFT entry, one reserved DVT entry (used to initialize Remote I/O Mapping); and, for each Mappable LU (used to establish maps to remote devices), a DVT entry.

Format

```
    IFT,/NS1000/DSREL/%ADV00,EIDV00,AL:DY,QU:FI,TX:2
    *
    DVT,,,LU:reservedLU,EDDV00,TX:0
    *
    DVT,,,LU:mappableLU,EDDV00,TX:5
    DVT,,,LU:mappableLU,EDDV00,TX:5
                  :
                  :
    DVT,,,LU:mappableLU,EDDV00,TX:5
```

Parameters

reservedLU A dummy LU that is otherwise unused.

mappableLU An LU to be used as a Mappable LU. Each *mappableLU* must be unique and otherwise unused. The *mappableLU*s can be used for resource sharing and from remote HP 1000 nodes to establish interactive sessions.

Note that the IFT entry does not specify a select code (SC), and that the reserved DVT entry has no extent words.

Example

```
IFT,/NS1000/DSREL/%ADV00,EIDV00,QU:FI,TX:2,AL:DY
*
DVT,,,LU:81,EDDV00,TX:0
DVT,,,LU:82,EDDV00,TX:5
```

X.25

X.25/1000 used with NS-ARPA/1000 requires the following table entries:

- one IFT entry per X.25 interface card
- two DVT entries per X.25 interface card
- one reserved IFT entry for `DDX00` driver
- two reserved DVT entries for communication with XNET
- two DVT entries per switched or permanent X.25 Virtual Circuit (user LUs).

Format

```
    *Interface card entries
    *
    IFT,/RTE_A/%IDS00,SC:selectcode,EIDS00,QU:PR,TX:12
    DVT,/X25/REL/DD*60.REL,,LU:transmitLU,TX:24,EDD.60
    DVT,/X25/REL/DD*60.REL,,LU:receiveLU,TX:24,EDD.60
    *
    *Reserved entries for communication with XNET
    *
    IFT,/X25/REL/DDX00.REL,EIDX00,TX:1
    DVT,,,LU:XNETWriteLU,EDDX00,TX:2
    DVT,,,LU:XNETReadLU,EDDX00,TX:2
    *
    *Virtual Circuit (user) LUs
    *
    DVT,,,LU:userWriteLU,EDDX00,TX:32
    DVT,,,LU:userReadLU,EDDX00,TX:32
    :
    :
    DVT,,,LU:userWriteLU,EDDX00,TX:32
    DVT,,,LU:userReadLU,EDDX00,TX:32
```

Parameters

selectCode	The select code of the X.25 interface card.
transmitLU *receiveLU*	Consecutive LUs for the interface card.
XNETWriteLU *XNETReadLU*	Reserved, consecutive LUs for communication with XNET.
userWriteLU *userReadLU*	Reserved, consecutive LUs for each X.25 Virtual Circuit. NS-ARPA/1000 uses the *userWriteLU*s.

Note that that you do not specify a select code for the `DDX00` IFT entry; this is because `DDX00` is a pseudo-interface driver.

The number of extent words for the user LUs is `32` (19 words for `DDX00` plus 13 for `%CXL66`, the NS-ARPA/1000 customizing subroutine).

For more information on X.25 generation requirements, refer to *DSN/X.25/1000 Reference Manual*.

Node Lists

Do not link any NS-ARPA/1000 DVTs with an RTE generator `NODE` *statement*. In an RTE-A Table Generation Phase, a “node” describes a specific relationship between the devices (DVTs) connected to an interface (IFT). These nodes are not related to NS-ARPA nodes.

Memory Allocation

This subsection describes the portions of the memory allocation phase that are affected by NS-ARPA/1000. For more detailed information about class numbers and resource numbers, refer to the subsection “System Resources” in Section 4, “Internal Resources.” Additional requirements are also needed for RTE-A or other subsystems.

Class Numbers

A sufficient number of class numbers for NS-ARPA/1000 is 30.

If it does not support DS/1000-IV Compatible Services, NS-ARPA/1000 requires a minimum of four class numbers. Otherwise, a minimum of seven class numbers is needed to support DS/1000-IV Compatible Services. Additional class numbers may be needed for such programs as PTOP master and slave programs, REMAT, RMOTE, and TELNET.

Resource Numbers

A sufficient number of resource numbers is 20.

ID Segments

60 to 80 ID segments are needed for NS-ARPA/1000 with DS, NS, ARPA, and BSD IPC services. Adjust the number according to the number of additional user and RTE programs in the system.

SAM and XSAM

32 pages of SAM are recommended; however, NS-ARPA can work with fewer pages of SAM.

The amount of XSAM is calculated as follows:

```
    (50 words * no. of concurrent TELNET user and server programs) + 2 pages
```

System Memory Block

NS-ARPA/1000 uses the System Memory Block (SMB) area to store some DS/1000-IV Compatible Services tables. If you configure the NSINIT default values for these tables, you will need 300 words for the SMB. The maximum amount of SMB required is 3312 words.

If you do not have DS/1000-IV Compatible Services, then you need to allocate at least one word of SMB.

Refer to the subsection “System Memory Block” in Section 4, “Internal Resources,” for more information.

Labeled Common

NS-ARPA/1000 does not support more than six pages of system common (includes both labeled and blank common). Of the six pages, one page of labeled common is used by NS-ARPA. Relocate the following NS-ARPA/1000 modules in labeled system common:

`RE,/NS1000/DSREL/%RESA` Required labeled common module

`RE,/X25/REL/#XCOM.REL` *Optional*. Relocate this X.25 module if you have X.25 links

`RE,/IMAGE/RDTB.REL` *Optional*. Relocate this remote Image module if you have Image/1000.

`RE,/RTE_A/HPMDM_TABLE.REL` *Optional*. Relocate this RTE module for the TELNET pseudo terminal LUs. Each TELNET pseudo terminal LU requires one `HPMDM_TABLE` table entry. The `HPMDM_TABLE` provided by HP allows up to eight LUs. If you wish to increase the number of LUs, you need to alter the file `/RTE_A/HPMDM_TABLE.MAC` before relocating `/RTE_A/HPMDM_TABLE.REL` in labeled system common. The pseudo terminal LUs are initialized by executing the RTE program, `HPMDM`. The above procedures are explained in the subsection, “TELNET Pseudo Terminal LU Initialization,” of the “Additional Initialization Tasks” section. Refer to this subsection first before proceeding with your system generation. `HPMDM` is also explained in RTE-A documentation.

You must search the following NS-ARPA/1000 library to resolve external references in `%RESA` as follows:

```
    SE,/NS1000/LIB/NSSYS.LIB
```

Default Library Specification

In the non-CDS library specification of the system generation, specify the following libraries, in the order listed below:

`LIB,BIGNS.LIB`	Merged NS-ARPA/1000 user library.
`LIB,SEC1000.LIB,,`	Security/1000 library.
`LIB,$FNDLB`	Required for NS-ARPA modules appended to user programs; also needed to link NFTMN.
`LIB,PASCAL.LIB`	Required for NS-ARPA modules appended to user programs.
`LIB,$BIGLB,,,,,,`	RTE-A library.

In the CDS library specification of the system generation, specify the following libraries, in the order listed below:

`LIB,BIGNS_CDS.LIB`	Merged NS-ARPA/1000 CDS user library.
`LIB,SEC1000CDS.LIB,,`	Security/1000 CDS library.
`LIB,SEC1000.LIB,,`	Security/1000 library.
`LIB,$FNDLB`	Required for NS-ARPA modules appended to user programs.
`LIB,PASCAL_CDS.LIB,,,`	Required for NS-ARPA modules appended to user programs.
`LIB,$BGCDS,,,,,,`	Required for NS-ARPA modules appended to user programs.
`LIB,$BIGLB,,,,,,`	RTE-A library.

The `BIGNS` and `BIGNS_CDS` libraries are built by the transfer file `/NS1000/CMD/INSTALL_NS1000.CMD` . Refer to Section 6, “NS-ARPA/1000 Program and File Installation,” for more information.

6

NS-ARPA/1000 Program and File Installation

This section describes the tasks you need to perform to install the NS-ARPA/1000 programs. The tasks you can execute from the BOOT command and system WELCOME files are described in Section 7, "BOOT and WELCOME Files."

Note If you have special application needs that require changes to the following areas from the system default, refer to Section 4, "Internal Resources," before installing NS-ARPA/1000:

- Memory used for NS-ARPA/1000
- System resources used for NS-ARPA/1000

The following example files for NS-ARPA/1000 installation are included with the NS-ARPA/1000 software:

`/NS1000/EXAMPLES/A91790_INSTALL.TXT`	software installation overview for NS-ARPA and DS
`/NS1000/EXAMPLES/A91790_EZ_INSTAL.TXT`	software installation overview for NS-ARPA and LAN
`/NS1000/EXAMPLES/#ANSNS`	sample system generation answer file
`/NS1000/CMD/INSTALL_NS1000.CMD`	program loading file
`/NS1000/EXAMPLES/NSSTART.CMD`	sample NS-ARPA start and initialization file
`/NS1000/EXAMPLES/NSSTART_EZ.CMD`	short version of `NSSTART.CMD`
`/NS1000/CMD/OF_NS.CMD`	sample command file for removing NS-ARPA programs

In addition, an example INETD configuration file (`/ETC/INETD.CONF`) and Services Mapping file (`/ETC/SERVICES`) are provided.

Blank worksheets, NSINIT examples, and an NRINIT example are in Appendix A, "Configuration Worksheets."

Installing NS-ARPA/1000 Programs

To load the NS-ARPA/1000 programs onto your system, HP provides the CI command file, /NS1000/CMD/INSTALL_NS1000.CMD . Read this file carefully.

The RTE-A master installation command file, /RTE_A/RTE_INSTALL.CMD can be used to invoke INSTALL_NS1000.CMD. Read the file RTE_INSTALL.CMD for information on its usage.

INSTALL_NS1000.CMD performs the following tasks:

- builds the NS-ARPA/1000 user libraries
- links the NS-ARPA/1000 programs
- copies the NS-ARPA/1000 libraries in /LIBRARIES
- copies the NS-ARPA/1000 message and help files into the appropriate directories

To run the CI transfer file, set your working directory to /NS1000 and transfer to /INSTALL_NS1000.CMD:

```
CI> WD /NS1000
CI> TR CMD/INSTALL_NS1000.CMD [Q]
```

where Q means select the quiet mode and bypass the interactive session. If Q is not specified, /INSTALL_NS1000.CMD prompts you for the following information:

- *Destination directories for files created during installation*. Directories are required for programs, system files, help files, library files, and map files.
- *Directory of RTE-A relocatable files*. Files in this directory are used when linking the programs, DSRTR, and TRFAS.
- *Snap file to use to link NS-ARPA programs*. Make sure that you use the system SNAP file of your *target* NS-ARPA system. The NS-ARPA/1000 programs must be linked with the system SNAP file of your target NS-ARPA system.
- *Access to a DS/3000 system or not*. If you want RTE-to-MPE DS backward compatible services, programs for DS/3000 access are linked and the user libraries configured accordingly. However, having libraries with DS/3000 access causes the BSD IPC routines to be excluded. Refer to “Building NS-ARPA Libraries” in the next subsection for detailed information.
- *Access to a DS/3000 system with X.25 or not*. If you will be using an X.25 connection to a DS/3000 system, X.25 routines will be included in the user libraries. The programs UPLIN, RQCNV, and RMOTE will be linked with the necessary X.25 routines.

Note DO NOT ALTER THE PRIORITIES OF ANY NS-ARPA/1000 SYSTEM PROGRAMS.

The priority of all the NS-ARPA/1000 programs must be higher than any user program which makes use of their capabilities. Be sure that your own programs do not have priorities higher than 30, that they are not in the range of 1 to 30. User programs with unnecessarily high priorities can delay necessary network processing and cause errors and/or poor performance.

Building NS-ARPA Libraries

A non-CDS library and a CDS library are created by the /NS1000/CMD/INSTALL_NS1000.CMD command file. These libraries are BIGNS.LIB and BIGNS_CDS.LIB, respectively. These libraries must be specified in the non-CDS and CDS library specifications of the system generation. Refer to Section 5, "System Generation."

These libraries are searched when linking NS-ARPA user programs (such as NetIPC, RPM, PTOP). These libraries are composites of libraries specified below. Do *not* rename the libraries to the names of existing libraries, such as NSLIB.LIB, NSLIB_CDS.LIB, NSSYS.LIB, or NSSYS_CDS.LIB.

The following libraries are merged into BIGNS.LIB or BIGNS_CDS.LIB:

- NSLIB.LIB and NSLIB_CDS.LIB are for NS-ARPA user programs (such as NetIPC, RPM, or PTOP).
- %POPN1 is the HP 1000-to-HP 1000 version of POPEN.
- DS3K.LIB contains the routines for RTE-MPE backward compatible services (access to a DS/3000 system).
- BSD_CDS.LIB contains the BSD IPC routines.

Programs that use the NS Common Services, must be compiled in CDS. Programs that use the DS/1000-IV Compatible Services (RTE-MPE) over X.25 links must be compiled in non-CDS only.

BSD IPC and DS/3000 Libraries

Because BSD_CDS.LIB and DS3K.LIB contain different subroutines with the entry point, FCNTL, they cannot be merged into the same library. To avoid conflict, only one of these libraries can be merged with the NSLIB libraries to create the BIGNS libraries. The CI transfer command file, /NS1000/INSTALL_NS1000.CMD , prompts you to choose either BSD_CDS.LIB or DS3K.LIB by using a CI variable, NS_DS3K.

If you create BIGNS_CDS.LIB without BSD_CDS.LIB, then you have to search BSD_CDS.LIB first when linking BSD IPC programs. This resolves external references for BSD IPC programs. Then BIGNS_CDS.LIB must be searched. This library is the default and is searched automatically by LINK.

Building BIGNS and BIGNS_CDS

How the BIGNS and BIGNS_CDS libraries are built depends on if you have X.25 links and if you are accessing a DS/3000 system. The libraries are built with one of these three possible configurations; the libraries are in the /NS1000/LIB directory:

- *Default with no DS/3000 access.*

```
BIGNS.LIB          = NSLIB.LIB + %POPN1
BIGNS_CDS.LIB      = BSD_CDS.LIB + NSLIB_CDS.LIB + %POPN1
```

- *With DS/3000 access and no X.25 link (*`BSD_CDS.LIB` *must be searched separately when linking a BSD IPC program).*

```
BIGNS.LIB          = DS3K.LIB + $D3N25 + NSLIB.LIB
BIGNS_CDS.LIB      = DS3K.LIB + $D3N25 + NSLIB_CDS.LIB
```

- *With DS/3000 access and with X.25 link. (*`BSD_CDS.LIB` *must be searched separately when linking a BSD IPC program).*

```
BIGNS.LIB          = DS3K.LIB + $D3X25 + NSLIB.LIB
                       + /X25/REL/XINXA.REL + /X25/REL/X25LB.LIB

BIGNS_CDS.LIB      = DS3K.LIB + $D3X25 + NSLIB_CDS.LIB
                       + /X25/REL/XINXA.REL + /X25/REL/X25LB.LIB
```

`/X25/REL/XINXA.REL` and `/X25/REL/X25LB.LIB` are in the X.25/1000 product.

Additional Loading Information

This subsection provides loading information that supplements the information in the CI command file `/NS1000/CMD/INSTALL_NS1000.CMD` .

DSAM Reserved Partition Size and Location

The size and location of the DSAM SHEMA partition, `#DSAM`, are determined when loading MMINIT, the Memory Manager routine. RTE allocates a partition that is one page larger than the SHEMA size specified in the `EM` command. For example, if you specify:

```
SH #DSAM
EM 150
```

RTE allocates a 151-page partition for `#DSAM` in dynamic memory. These defaults are in the LINK `/NS1000/LOD/MMINIT.LOD` file used by the command file, `/NS1000/CMD/INSTALL_NS1000.CMD` . If you have no reason to change these defaults, skip the rest of this subsection. In most cases, the DSAM SHEMA size of 150 pages specified in the `/NS1000/LOD/MMINIT.LOD` file is adequate. However, if you need to increase or decrease DSAM's partition size, refer to Section 4, "Internal Resources."

By default, `#DSAM` is not assigned to a reserved partition. To assign `#DSAM` to a reserved partition, you must create a reserved partition by using the `RV` command in your BOOT command file. Each `RV` and `AS` (assign partition) command creates one reserved partition. The first partition created is partition number 1 and so on. Refer also to the "BOOT Command File" and "DSAM Reserved Partition Assignment" subsections in Section 7.

Next, assign `#DSAM` to the reserved partition when MMINIT is linked. To assign MMINIT's SHEMA, `#DSAM`, to a partition or to change the partition size, modify the LINK commands `SH` and `EM` when linking MMINIT. You can modify the following two LINK commands in either the `/NS1000/LOD/MMINIT.LOD` file or after the LK command when relinking MMINIT.RUN.

```
    SH #DSAM partNum
    EM SHEMA_size
```

The *`partNum`* is the reserved partition's number. The value for *`partNum`* must match the number of the partition reserved for DSAM, as specified in the BOOT command file.

The *`SHEMA_size`* is the size of the partition, in pages. The value for *`SHEMA size`* must be less than the size of the reserved partition specified in the system BOOT command file. Remember, RTE allocates one extra page in the partition.

Next, load `MMINIT.REL` with the modified `MMINIT.LOD` file.

```
    CI> WD /NS1000
    CI> LINK PROGRAMS/MMINIT.RUN LOD/MMINIT.LOD
```

Or you can relink as follows:

```
    CI> WD /NS1000
    CI> LINK PROGRAMS/MMINIT.RUN
    link version 6100       Use ? for help
    link: LK
     Ready to modify MMINIT.RUN::PROGRAMS
     Use LO to examine programs
    link: LO
     Program MMINIT.RUN::PROGRAMS
      shareable              = No
      priority               = 99
      SHEMA name             = #DSAM
      SHEMA partition        = None
      SHEMA partition size   = 151
      EMA size               = 150 pages
      mseg                   = 23 pages
      partition #            = None
      program size           = 7 pages
      partition size         = 7 pages
      Program capability     = 31
      Required capability    = 0
      Debug monitor mode     = OFF
    link: SH,#DSAM,1            Assign SHEMA #DSAM to partition 1
    link: EM,200                Set EMA size to 200 pages; SHEMA partition will be 201 pages
    link: LO
     Program MMINIT.RUN::PROGRAMS
      shareable              = No
      priority               = 99
      SHEMA name             = #DSAM
      SHEMA partition        = 1
      SHEMA partition size   = 201
      EMA size               = 200 pages
      mseg                   = 23 pages
```

```
  partition #          = None
  program size   =      7 pages
  partition size =      7 pages
  Program capability    = 31
  Required capability   = 0
  Debug monitor mode    = OFF
link: EN                 Update program file and exit LINK
 Updating program file MMINIT.RUN::PROGRAMS
```

Remote File Access

Each file concurrently open at the local node from remote nodes for DS/1000-IV Compatible Services requires a Remote File Access Monitor (RFAM) Data Control Block (DCB). Each DCB requires 153 words of RFAM program space. RFAM allocates DCBs from its internal buffer and from the remainder of its program partition. RFAM should be assigned to a reserved partition. You can size up RFAM's program partition to accommodate more DCBs. To do this, you can use the LINK or RTE CI `SZ` commands.

NSTRC and FMTRC VMA Sizes

You must load NSTRC and FMTRC with the same VMA sizes. The default values in the load files in `/NS1000/CMD/INSTALL_NS1000.CMD` use a working set of 32 pages and a VMA size of 128 pages.

The VMA size determines the file size for trace files, that is, how many trace records can be logged into a trace file. The working set determines how much is in memory when NSTRC runs. When the trace file is full, the logged trace records are overwritten with more recent trace records. If the overwriting occurs too frequently, you may want to increase the VMA size. If NSTRC starts to drop trace records, then page faults must be occurring. The VMA size and working set size need to match more closely in size.

To change either the VMA size or working set size, relink NSTRC and FMTRC. Another way is to `RP` the programs, and use the RTE `VS` command and/or RTE `WS` command.

NFTMN

NFTMN must be loaded with the library $FNDLB even if the library, $FDSLB, is used.

NS-ARPA/1000 Program and File Directories

The NS-ARPA/1000 programs and files should be installed in the directories listed below. This is done for you if you use the file `/NS1000/CMD/INSTALL_NS1000.CMD` with the default values. In the list below, the programs preceded with an asterisk (`*`) are installed only if HP 3000 access is requested.

```
/PROGRAMS
   BREVL.RUN
   BRTRC.RUN
   CONSM.RUN
   DSCOPY.RUN
   DSMOD.RUN
   DSVCP.RUN
   FMTRC.RUN
   FSRV.RUN
   FTP.RUN
   FTPSV.RUN
   LOGCHG.RUN
 * LOG3K.RUN
   NRLIST.RUN     (Optional)
   NSINF.RUN
   NSTRC.RUN
   PING.RUN
   PRODC.RUN
   PRDC1.RUN
   RDATE.RUN
   REMAT.RUN
 * RMOTE.RUN
   TELNET.RUN
   TNSRV.RUN
 * TRC3K.RUN

/SYSTEM
   DSCOPY.HLP
   FTP.HLP
   TELNET.HLP     (Optional)
   NSERRS.MSG
   NSINIT.MSG

/LIBRARIES
   BIGNS.LIB
   BIGNS_CDS.LIB
 * BSD_CDS.LIB
   ERRNODEC.REL
```

```
/PROGRAMS
   INPRO.RUN
   MMINIT.RUN
   NSINIT.RUN
   OUTPRO.RUN
   UPLIN.RUN

/HELP
   BREVL
   BRTRC
   DSCOPY
   EVMON
   EXPORTS
   FMTRC
   FSRV
   FTP
   INETD
   LOGCHG
   NRLIST
   NSTRC
   PING
   RDATE
   TELNET
   TZTAB

/ETC
   INETD.CONF
   SERVICES
   TZTAB

/CATALOGS
   INETD.C000
```

```
/PROGRAMS
   #SEND.RUN
   APLDR.RUN
 * CNSLM.RUN
   DLIST.RUN
 * DSLIN.RUN
   DSRTR.RUN
 * DSTES.RUN
   EVMON.RUN
   EXECM.RUN
   EXECW.RUN
   GRPM.RUN
   IFPM.RUN
   INETD.RUN
   IOMAP.RUN
   LUMAP.RUN
   LUQUE.RUN
   MATIC.RUN
 * MVCP3.RUN      (Optional)
   NFTMN.RUN
   NRINIT.RUN
   OPERM.RUN
   PROGL.RUN
   PTOPM.RUN
   QCLM.RUN
   QUEUE.RUN
 * QUEX.RUN
 * QUEZ.RUN
   RFAM.RUN
 * RPCNV.RUN
   RPMMN.RUN
 * RQCNV.RUN
   SYSAT.RUN
   TRFAS.RUN
   VCPMN.RUN
```

For more information on which NS-ARPA/1000 programs and files to restore (RP), refer to Section 4, “Internal Resources.” Restore programs either in the BOOT command file or WELCOME file.

91750A DS/1000-IV Program Files

NS-ARPA/1000 program files are incompatible with DS/1000-IV (91750A) program files. If you have any DS/1000-IV program files, you *must keep these files in a separate directory*.

If you are installing a link other than a LAN link and/or need DS/1000-IV Compatible Services, read the file `/NS1000/EXAMPLES/A91790_INSTALL.TXT`. This file contains a summary of NS-ARPA/1000 installation procedures for such a configuration.

Related Subsystems

If you are installing a LAN link and do not need any other links and do not need DS/1000-IV Compatible Services, read the file `/NS1000/EXAMPLES/A91790_EZ_INSTAL.TXT`. This file contains a summary of fast, easy-to-use NS-ARPA/1000 installation procedures for such a configuration. These procedures are helpful for cross-system connectivity, because such connections to other types of HP computers (HP 9000, HP 3000) are on a LAN.

For LAN/1000, HP recommends that you load the LAN/1000 Node Manager. The Node Manager is used as a troubleshooting tool. Refer to the *HP 12076A LAN/1000 Link Node Manager's Manual*, part number 12076-90002.

If you are using any of the following subsystems, you must relink the listed modules each time you install a new version of NS-ARPA/1000 to ensure compatibility (refer to Table 6-1).

Table 6-1. Related Subsystem Modules

Subsystem	Module	Comments
Transparent File Access	DSRTR	Required at the node at which the file access request originates. DSRTR is linked by the command file, `/NS1000/INSTALL_NS1000.CMD`
	TRFAS	Required at the node that is to service the file access request. TRFAS is linked by the command file, `/NS1000/INSTALL_NS1000.CMD`
Remote Image/1000	RDBAM	Remote Database Access Monitor. Search `NSSYS.LIB` and `NSLIB.LIB.`

7

BOOT and WELCOME Files

This section describes the tasks that you can execute from the BOOT command file and system WELCOME file.

The command files, `/NS1000/EXAMPLES/NSSTART.CMD` and `/NS1000/CMD/NSSTART_EZ.CMD` restore programs and start the execution of LAN/1000 and NS-ARPA/1000 programs. Refer to these files. The commands from these files may be inserted into your BOOT and/or WELCOME files.

BOOT Command File

Modify the BOOT file as described in the steps below:

1. Restore NS-ARPA/1000 programs (RTE `RP` command). The `RP` commands can be done in the WELCOME file instead. Refer to the `/NS1000/EXAMPLES/NSSTART.CMD` file to determine which commands to add to your BOOT file. Review the `RP` commands and delete the commands for NS-ARPA/1000 programs you will not use. The `RP` commands for programs related to DS/1000-to-DS/3000 services, DSVCP (remote VCP), and PROGL (remote system download) are commented out with asterisks (`*`). Remove the asterisks to use the `RP` commands for these programs.

2. (Optional) Reserve a partition (RTE `RV` command) for the NS-ARPA SHEMA `#DSAM`.

   ```
   RV,151
   ```

   Be sure to re-link MMINIT to assign `#DSAM` to this reserved partition.

   ```
   CI> LINK MMINIT.RUN
          LK
          EM size
          SH #DSAM partition_num
          EN
   ```

3. (Optional) Assign programs to reserved partitions (RTE `AS` commands).

DSAM Reserved Partition Assignment

At boot time, you would reserve the partition for DSAM by entering the following command in the BOOT command file:

```
RV,numPages
```

This command defines a reserved partition with *numPages* pages, and with a partition number that is determined by the command's position in the BOOT file. RTE assigns partition numbers sequentially, starting from 1, with each `RV` or `AS` command. The value for *numPages* must be at least one page larger than the SHEMA size specified by the LINK `EM` command when MMINIT was linked. MMINIT and DSAM are described in more detail previously in Section 6 in the subsection, "DSAM Reserved Partition Size and Location."

Assigning Other Partitions

HP recommends that you restore (`RP`) and assign partitions for the following program:

```
UPLIN
```

If the local node has any Router/1000 (RTR) or Bisync or X.25 links to HP 3000 nodes, HP recommends that you `RP` and assign a partition for the following program:

```
QUEUE
```

If the local node has any X.25 links, HP recommends that you restore (`RP`) and assign a partition for the following program:

```
XNET
```

Refer to the X.25 manuals for complete information about X.25 links.

For example, if you have a LAN link and an HDLC (RTR) link, you could include the following commands in your Boot command file:

```
RP,UPLIN
RP,QUEUE
  :
  :
END
RV,151
AS,UPLIN
AS,QUEUE
  :
  :
```

You can restore (`RP`) the remaining NS-ARPA monitor programs required for your system in the boot or welcome file. For a list of the NS-ARPA programs required for your systems, refer to Section 4, "Internal Resources."

Note The following programs should not be restored in a multiuser system. These programs will be cloned for each user.

NFT server programs: PRODC, PRDC1, and CONSM
FTP server program, FTPSV
TELNET server program, TNSRV

WELCOME File

Modify the WELCOME file as described in the steps below:

1. Restore any NS-ARPA monitor programs not `RP`ed from the BOOT command files. For a list of the NS-ARPA programs required for your systems, refer to Section 4, “Internal Resources.”

2. Initialize the link subsystems (such as LAN/1000 and/or X.25/1000). Refer to Section 8, “Link Initialization.”

3. Initialize NS-ARPA by running NSINIT. Refer to Section 9, “NSINIT, NS-ARPA/1000 Initialization.”

4. (Optional) Initialize the Nodal Registry by running NRINIT. Refer to Section 11, “NRINIT, Nodal Registry Configuration.”

   You do *not* have to initialize the Nodal Registry if all the nodes on the network are HP systems and have an IEEE 802.3 LAN link.

5. (Optional) Initialize the following NS-ARPA services: TELNET pseudo terminal LUs, Remote I/O Mapping, and Bisync links to HP 3000 computers. Refer to Section 12, “Additional Initialization Tasks.”

6. If you want to use the RMOTE `MO` command, you must install the slave program, `COPY3K.PUB.SYS`, on the HP 3000 after you have initialized NS-ARPA. Refer to Section 12, “Additional Initialization Tasks.”

Reboot your system to execute the revised BOOT and WELCOME files.

7. (Optional) Run FSRV.

8

Link Initialization

If you have X.25 or LAN/1000 links, you must initialize those links before initializing NS-ARPA/1000.

- For X.25 links, refer to *DSN/X.25/1000 Reference Manual*.
- For LAN/1000 links, refer to the *HP 12076A LAN/1000 LANIC Link Installation Manual* and *HP 12076A LAN/1000 Link Node Manager's Manual*. Initialize the LAN/1000 driver for each LANIC card with an RTE `CN` command or `EXEC` call.

In addition, the following subsection describes what you must do to use LAN/1000 with NS-ARPA/1000.

LAN/1000 Links

Note If you have purchased the 12079A LAN/1000 Direct Driver Access (DDA) product, please note that HP does not support NS-ARPA/1000 under the following condition:

- NS-ARPA/1000 used with an altered LAN/1000 interface driver.

With RTE-A and NS-ARPA/1000 software revision codes of 5.0 or later, the 12079A DDA product is supported when NS-ARPA/1000 is also installed. There are restrictions applied to user-written LAN programs. Refer to the *HP 12079A LAN/1000 Link Direct Driver Access Manual* for more information.

To use LAN/1000 with NS-ARPA/1000, you must perform the following tasks in addition to the tasks described in the LAN/1000 manual set:

- If you have LAN/1000 Node Manager software and the appropriate multicast address file for your card (the file `/files802/addr`*`hhhhhhhhhhhh`*`/mcast.txt`, where *`hhhhhhhhhhhh`* is the card's station address) on your system, *the multicast addresses for NS-ARPA must be included in the multicast address file.*

  Note that you do not need a multicast address file for your card if the NS-ARPA/1000 multicast addresses are the only multicast addresses used.

In addition, HP *recommends* that you perform the following task:

- Use the default LAN station addresses (the address shipped with the card) and the NS-ARPA/1000 Probe default multicast addresses (`09-00-09-00-00-01` and `09-00-09-00-00-02`).

For more information, refer to the *LAN Cable and Accessories Installation Manual*, *HP 12076A LAN/1000 Link LANIC Installation Manual*, and *HP 12076A LAN/1000 Node Manager's Manual*.

9

NS-ARPA/1000 Initialization

NS-ARPA/1000 Initialization Overview

After you have generated NS-ARPA/1000 in your system and started your system, you can initialize NS-ARPA/1000 by running NSINIT. NSINIT initializes NS-ARPA/1000 by allocating internal resources, setting NS-ARPA parameters, scheduling services and monitors, and enabling LUs.

Note Before attempting to initialize your node, you must read the sections “Overview” and “Network Configuration Planning,” and you should fill out a Network Worksheet.

You must keep an up-to-date copy of the NSINIT answer file, either on paper or disc. Should you have any problems with your system, HP field support will require the NSINIT answer file.

Before you schedule NSINIT, you should be sure that the necessary NS-ARPA/1000 files and programs are restored or are in the appropriate directory. The required programs and files are listed in Section 6, “NS-ARPA/1000 Program and File Installation,” and described in Section 4, “Internal Resources.”

You must have superuser capability to run NSINIT.

NSINIT uses a dialogue to prompt the user for initialization information. Use the configuration information recorded on your network worksheet to answer questions for NSINIT. Refer to Section 3, “Network Configuration Planning,” for information about configuring your node and network, and to Appendix A for a sample network worksheet. You can specify an initialization answer file that contains the responses to the dialogue or you can respond to the dialogue interactively.

In addition, NSINIT can create answer files. An *answer file* contains the answers to the questions in the NSINIT dialogue and is created by NSINIT by recording your responses to its questions. Once you have created an answer file, you can run NSINIT using the answer file to provide the responses. You can schedule NSINIT from the WELCOME file using the initialization answer file. Sample answer files are shown in Appendix A and are included in the `/NS1000/EXAMPLES` directory.

To help you read and modify answer files, NSINIT also prints the NSINIT questions in these files, which are treated as comments. Comment lines are described in the "Comments" subsection later in this section. You can create NSINIT answer files for use at other nodes. In some cases, you can also edit NSINIT answer files to reflect changes in the network. Information on editing NSINIT answer files is provided at the end of this section.

Typically for the first time user, NSINIT is run twice. First, NSINIT is run interactively and an answer file is created. The start up response is NO, so that NS-ARPA/1000 is not initialized. Then the answer file is reviewed against the network worksheet and possibly revised. If you are satisfied with the NSINIT answer file, then run NSINIT a second time with the answer file as input and the start up response is YES.

For an experienced user, NSINIT is usually scheduled in the WELCOME file with a specified answer file. Scheduling NSINIT in the WELCOME file can automate the initialization process.

NSINIT Runstring

```
[RU,]NSINIT[,inputDevice][,outputDevice][,logDevice]
```

Parameters

inputDevice Device (file or interactive device LU) that provides input for NSINIT dialogue. The *inputDevice* can be an answer file previously created by running NSINIT. If *inputDevice* is an interactive device, NSINIT will also print prompts and error messages to the device.

Default: local LU 1 (scheduling terminal).

outputDevice Device (file name or device LU) to which NSINIT will write the dialogue. If this is a file, it can later be used as an answer file. If *outputDevice* and *inputDevice* are files, they cannot be the same file. If you enter the name of an existing file, NSINIT will ask if you want to overwrite the file. If you enter LU 0, NSINIT does not create any output.

Default: If *inputDevice* is interactive, NSINIT will prompt you for an output file name.

If *inputDevice* is not interactive, the default is LU 0 (no output). During an interactive NSINIT dialogue, NSINIT prompts you again for an output file name. Refer to "NSINIT Dialogue" later in this section.

logDevice Device (file name or device LU) to which NSINIT will log any errors.

Default: local LU 1 (scheduling terminal).

Error Handling

If you enter an inappropriate response during an interactive NSINIT session, NSINIT will generally print an error message and re-prompt you until you enter an acceptable response.

If NSINIT cannot recover gracefully from the error, NSINIT will abort.

If you were also building an answer file, NSINIT writes the incorrect response to the answer file and prints a message warning you that you must edit your answer file before using it. If NSINIT can recover from the error, NSINIT will replace the inappropriate response in the answer file with your acceptable response.

If you enter an inappropriate response while using an answer file as input, NSINIT will print an error message to `logDevice` and terminate.

If NSINIT terminates with an error, it will return an error code to its scheduling program in PRTN parameter $RETURN1; otherwise, it will return a zero. For information on PRTN parameters, refer to the *RTE-A User's Manual* and *RTE-A Programmer's Reference Manual*. NSINIT error messages are documented in the *NS-ARPA/1000 Error Messages and Recovery Manual*.

Dialogue Syntax

The NSINIT dialogue uses the following syntax:

- Square brackets ([ ]) enclose acceptable responses to a question or contain optional parameters.
- Parentheses further enclose the the default response to a question.
- Slashes (/) separate items in a list of acceptable responses; enter one item. For example,

  ```
  Do you want to modify these
  values [Y/(N)]?
  ```

  The acceptable responses are Y and N. The default response is N.

- Ellipses separate items in a range of acceptable responses. Unless otherwise noted, the acceptable responses in a range are integers, and the range boundaries are inclusive. An upper bound of nnn indicates infinity. For example,

  ```
  Enter the default buffer size in
  bytes [700..(2048)]
  ```

  The acceptable responses are integers in the range from 700 to 2048, inclusive. The default response is 2048.

- Angle brackets (< >) separate fields when the format for responses is given. Do not enter the angle brackets. Use commas to delimit fields in your response; if there is a default value for a field, you can omit the field but you must enter a comma placeholder if you enter any subsequent fields.

  If the field is in lower case, it is a description of the entry. If the field is in upper case, it is a literal field; enter it as it is written. A literal field may have a list of acceptable responses separated by slashes or a range of acceptable responses. For example

  ```
  <local IP addr>,[<subnet mask>,]
  RTR, <segsize: [1200..(8000)]>
  ```

  An acceptable response would be an IP address, followed by a subnet mask, the literal field RTR, and an integer between 1200 and 8000. If the subnet mask is not used, the <local IP addr> is immediately followed by the literal field RTR.

NSINIT Dialogue Responses

Besides replying to NSINIT questions with specific values, you can enter the following responses. These responses are also shown in the NSINIT dialogue.

RETURN or /D — Use the default response to the question. If there is no default, NSINIT will print an error message. If you are creating an answer file, NSINIT replaces RETURN with /D.

AB or /A — Abort NSINIT. NSINIT closes its files and terminates. NSINIT will write the last prompt to the output file, but not the AB or /A.

`/E`	End data entry. The `/E` response is only used for questions with multiple responses. If entered at any other time, NSINIT prints an error message.

Parsing

NSINIT upshifts any lower-case entries.

If you are creating an answer file with a text editor, you can use blank lines to indicate the default answers (hit carriage return). However for clarity, HP recommends that you use `/D` to indicate the default.

Comments

NSINIT treats any text on a line following an asterisk as a comment. When NSINIT creates an answer file, the NSINIT questions are written to the file as comments.

Local Node

In the explanatory text for the NSINIT dialogue, *local node* refers to the node that you are creating an answer file for, which need not be the node at which you are running NSINIT.

NSINIT Dialogue

This subsection describes the NSINIT dialogue. The dialogue is shown in bold type.

Operation Mode

```
NS Initialization Program. Enter values as directed.
To use the default value enter /D or <CR>. To abort enter /A or AB.
* Network Initialization Options:
*       1: Build Output File.
*       2: Build Output File & Initialize Network.
*       3: Initialize Network.
*       4: Shut Down Network Subsystem.
* Enter an option number [(1)..4]:
```

Enter the appropriate option number (default is 1).

If NSINIT is preparing an output (answer) file, it will write `3` to the answer file here. This allows you to use the answer file to initialize NS-ARPA.

If you enter `2` or `3`, NSINIT checks that NS-ARPA/1000 is shut down, that the program name is `NSINI`, and that NSINIT was loaded as a system utility.

If you enter `4`, NSINIT skips to “Shut Down Dialogue.”

If you enter `1`, `2` or `3`, NSINIT will attempt to open the *outputDevice* if specified in the runstring.

NSINIT asks the next question only if you are running NSINIT interactively, entered option `1` or `2`, and you did not specify a file name for *outputDevice* in the NSINIT run string.

```
*Enter a file name for the output file [ /D = nsoutxx ]:
```

Enter a valid RTE file name for the output file. If NSINIT’s input device is a file, the output file must not be the same file. If you enter the name of an existing file, NSINIT will ask if you want to overwrite the file.

After NSINIT opens the file, it prints the following message:

NSINIT opened output file name: ***xxxx***

Default: `NSOUT`*nn* where *nn* is 1 to 99 and is incremented as necessary to form unique file names when NSINIT is scheduled. If there are 99 `NSOUT`*nn* files in the current working directory, NSINIT prints an error message and terminates.

Network Management

```
* Enter the local node name. Format: name.domain.organization
* (Each field may be 1..16 char)
*Local Name:
```

Enter the location name for this node. Each field, `name`, `domain`, and `organization`, must be 1 to 16 alphanumeric, underscore (_), or dash (-) characters; the first character must be alphabetic. You must delimit each field with a period. Refer to the section "Network Configuration Planning," subsection "Node Names" for more information. The local node is automatically put in the Nodal Registry. The Nodal Registry is further explained in the section, "Nodal Registry Configuration Planning," later in this manual.

Logging

The NS-ARPA event logging program, EVMON, records NS-ARPA/1000 *events*, and can help you if you have subsystem problems. For more information, refer to *NS-ARPA/1000 Maintenance and Principles of Operation*, section "NS-ARPA Event Logging."

```
*++ Classes of Events to log:
*       6: Resource limit exceeded
*       5: Disaster (irrecoverable error)
*       4: Error    (severe, but recoverable error)
*       3: Warning  (unexpected event )
*       2: Event message
*       1: Internal State Information
*
* Enter the event classes to log, one per line.
* [/D = 4, 5, and 6.] Type /E to end.
*  Event class:
```

The NS-ARPA Event Logger records the following classes of events:

Resource limit exceeded (class 6), which indicate that an NS-ARPA resource, such as DSAM, was depleted.

Disasters (class 5), which are catastrophic events that may cause NS-ARPA to suspend your system.

Errors (class 4), which are severe but recoverable errors. These errors are documented in *NS-ARPA/1000 Error Message and Recovery Manual*, part number 91790-90045.

Warnings (class 3), which indicate unexpected events

Event messages (class 2), which are internal messages passed between protocol handlers to handle events. Logging this information may produce a high number of logging entries and degrade system performance. HP recommends that you do not log event messages unless it is advised by your HP representative.

Internal State Information (class 1), which gives the status of internal modules (protocol-specific information). Logging this information may produce a high number of logging entries and degrade system performance. HP recommends that you do not log Internal State Information unless it is advised by your HP representative.

Enter the class or classes of events that you want NS-ARPA to log, one at a time.

If you do not enter any event class, NSINIT skips the next question. Otherwise, NSINIT prompts:

```
* Enter an event log file name [default = /system/ns_event.log]:
```

Enter the file name or LU to which you want NS-ARPA errors logged. If you do not fully-qualify the file name, the directory will default to the system session's default working directory.

Tracing

The NS-ARPA trace program, NSTRC, records messages, including data, as they are sent by NetIPC via a socket and as they are physically transmitted by an interface card through the network. NSTRC records these messages in binary form, to a file or LU. You can then use FMTRC to format the file.

```
* Do you want to start NSTRC when NS is enabled [Y/(N)] ?
```

Enter `Y` (YES) if you want NSINIT to attempt to schedule NSTRC immediately after NSINIT completes NS-ARPA initialization. Doing so will allow you to troubleshoot the network initialization process. NSTRC must be present and restored as a program file (via the `RP` command).

If you do not have to troubleshoot network startup, enter `N` (NO). Refer to *NS-ARPA/1000 Maintenance and Principles of Operation*, section "NS-ARPA Message Tracing," for more information on NSTRC.

Network File Transfer

NFT allows you to copy files from one node to another.

```
* Do you want Network File Transfer (NFT) [(Y)/N]?
```

If you enter `Y` (YES), NSINIT schedules NFTMN, the NFT monitor, at initialization time and increase NSINIT's default number of active system sockets. NSINIT also attempts to move the NFT error message file `/NS1000/NSERRS.MSG` to the DSAM SHEMA. If NSINIT is successful, the message file adds 3500 words to the space required for tables in DSAM.

If you enter N (NO), NSINIT skips to "DS/1000-IV Compatible Services." Otherwise, NSINIT displays the NFT default options and asks if you want to change any of them:

```
*++   Network File Transfer (NFT) ++*
*    Default Buffer Size:       2048 bytes
*    Transport Checksum Used:   No
*
* Do you want to modify these values [Y/(N)]?
```

Enter Y (YES) to modify NFT parameters; these parameters are used for NFT file transfers when the source file is at the local node. Otherwise, enter N (NO). You *must* enter Y (YES) if you want to transfer files to NS for the DEC VAX systems.

If you enter N (NO), NSINIT skips to "DS/1000-IV Compatible Services." Otherwise, NSINIT prompts:

```
* Enter the default buffer size in bytes [700..(2048)..4096]:
```

The NFT buffer size determines the size of the buffers that NFT will use for transmission via NetIPC. The buffers may be further segmented by the interface card or IP. When NFT initiates a logical connection for a file transfer, it negotiates the NFT buffer size with its peer at the other node. The NFTs exchange default NFT buffer sizes; if they do not match, the lower buffer size is used. For the best performance over LAN links, enter a multiple of 1450 (such as 2900). For the best performance over other links, enter the largest buffer size. Enter the NFT buffer size (bytes) that you want. If you want to transfer files to NS for the DEC VAX systems, the NFT buffer size must not be greater than 1423 bytes. If a larger buffer size is specified, file transfers to NS for the DEC VAX systems will fail and you will receive an NS/NFTERR 27 error.

Internal Resources: NFT buffers require DSAM.

```
* Should NFT always use transport checksumming [Y/(N)]:
```

When NFT initiates a logical connection for a file transfer, the Layer 4 protocol negotiates whether or not to use checksumming with its peer at the remote node. If either node wants to use checksumming, checksumming is performed. The Layer 4 checksum is performed in *addition* to the Layer 2 checksum. If Layer 4 performs checksumming, increased overhead is required.

If you enter Y, checksumming will always be performed.

If you enter N, checksumming will be performed only if the remote node requests it.

DS/1000-IV Compatible Services

The DS/1000-IV Compatible Services are: RTE Transparent File Access (TRFAS), Remote Operator Requests (REMAT), Remote Directory List (REMAT), Remote 3000 Session Access (RMOTE), Distributed EXEC (DEXEC), Program-to-Program Communication (PTOP), Remote Data Base Access (RDBA), Remote File Access (RFA), Remote I/O Mapping, and DS Virtual Control Panel (DSVCP).

```
*++   DS/1000-IV Compatible Services     ++*
*
* Do you want any DS/1000-IV compatible services [Y/(N)]?
```

Enter Y (YES) if you want to use DS/1000-IV Compatible Services at this node.

If you enter N (NO), NSINIT skips to "NS-ARPA Nodal Information." Otherwise, NSINIT prompts:

```
* Enter the maximum number of concurrent DS/1000-IV transactions on this
* node [1..(20)..100]:
```

Each programmatic or remote request that uses DS/1000-IV Compatible Services (such as REMAT, RMOTE, DLGON, HELLO, Remote I/O Mapping) is a DS/1000-IV transaction. Enter the maximum number of DS/1000-IV transactions that will be active at the local node at any time.

Internal Resources: Each DS/1000-IV transaction requires a transaction control block (TCB). TCBs used for transactions to HP 3000s are entered in the Process Number List (PNL).

```
* Enter the maximum number of concurrent DS/1000-IV remote transactions to
* access this node [1..(7)..253]:
```

Enter the maximum number of concurrent PTOPM or EXECW schedule requests from a remote node to the local node.

Internal Resources: This is the number of entries created in the POOL table.

```
*+++  DS/1000-IV Timeout Values  +++*
* Master timeout in seconds:      45
* Slave timeout in seconds:       30
* Idle "session" timer in hours:   5
*
* Do you want to modify these values [Y/(N)]?
```

Enter Y (YES) to modify these values; otherwise, enter N (NO).

If you enter N (NO), NSINIT skips the next three questions. Otherwise, NSINIT prompts:

```
* Enter the master timeout in seconds [5..(45)..1275]:
```

Enter the master timeout value. This value is the number of seconds that UPLIN will wait before timing out an unsatisfied master request initiated by the local node. If UPLIN times out a master request, UPLIN returns any system resources allocated for the request and logs the event.

The master timeout value must be greater than or equal to the MA retry limit (configured in the DS/1000-IV Compatible Transport section of the dialogue) times the MA timeout value (configured in the DS/1000-IV Compatible Transport section of the dialogue).

```
* Enter the slave timeout in seconds [5..(30)..1275]:
```

Enter the slave timeout value. This value is the number of seconds that UPLIN will wait before timing out an unsatisfied, inbound slave request at the local node.

```
* Enter the idle "session" timeout in hours [0..(5)..45]:
```

Enter the idle session timeout value. This value is the number of hours that UPLIN will wait before logging off an idle remote logon to the local node. `0` specifies an infinite timeout.

```
* Do you want any DS/1000-IV 1000-1000 compatible services [(Y)/N]?
```

Enter `Y` (YES) if you want to use any DS/1000-IV Compatible Services (RTE-RTE). If you enter `Y`, NSINIT will attempt to schedule IFPM (Interface Process Monitor) at initialization time.

```
* Do you want DS/1000-IV 1000-3000 services [Y/(N)]?
```

Enter `Y` (YES) if you want to use RMOTE or RTE-MPE RFA, PTOP, or any of the DS/1000-IV RTE-MPE utilities. If you enter `N`, NSINIT skips the next question. If you enter `Y`, NSINIT will attempt to schedule QUEX, QUEZ, RQCNV, and RPCNV at initialization time and prompt you with the following question:

```
* Enter the maximum number of concurrent DS/3000 users to access
* this node [0..(4)..10]?
```

Enter the maximum number of concurrent NS/3000 and DS/3000 users to access this node to use DS/1000-IV Compatible Services (1000-3000).

Internal Resources: This number is the number of entries created in the Transaction Status Table (TST).

```
* Should NSINIT try to schedule all the DS/1000-IV Monitors (VCPMN, TRFAS
* RDBAM, RFAM, PROGL, PTOPM, OPERM, EXECW, EXECM, DLIST, CNSLM) [(Y)/N]?
```

(NSINIT only lists CNSLM if you responded `Y` to the question `Do you want DS/1000-IV-DS/3000 services?`)

Refer to the section, "Internal Resources," subsection "Software Modules," for information on each monitor.

If you enter `Y` (YES), NSINIT will attempt to schedule all of the listed slave monitors, and skips the next question.

Otherwise, NSINIT prompts for a monitor name:

```
* Enter a Monitor name. Type /E to end:
```

Enter the name of each monitor that you want to schedule, one at a time. Enter `/E` to end data entry.

If you enter the name of a non-existent monitor, NSINIT will write the following message to the log device:

```
 ** (351) NSINIT:  Invalid monitor name. **
```

If you specified that you want NSINIT to schedule RFAM (explicitly or implicitly), RFAM will print a message at NS-ARPA initialization time telling you the maximum number of files that can be simultaneously open at the local node from remote nodes. RFAM must allocate the same number of Data Control Blocks (DCBs).

NS-ARPA Nodal Information

The questions in this section are for resources for NS-ARPA programs. NS-ARPA programs are those that provide NS-ARPA services *excluding* DS/1000-IV Compatible Services.

```
*++   NS Nodal Information    ++*
*
* Defaults are derived from previous responses.
*       Maximum number of active NS programs is:   nnn.
*       Maximum number of active NS sockets is:    nnn.
*
* Do you want to modify these values [Y/(N)]?
```

Enter `Y` (YES) to modify Nodal Resources; otherwise, enter `N` (NO).

If you enter `N`, (NO), NSINIT skips to “Nodal Registry.” Otherwise, NSINIT prompts:

```
* Enter the maximum number of active NS programs [1..(nnn)..100]:
```

The default maximum number of active NS-ARPA programs is determined by the following formula:

`number of NS-ARPA subsystem programs + number of user programs`

The *`number of NS-ARPA subsystem programs`* is 6 if you have NFT, 5 if you do not. The *`number of user programs`* is the maximum number of user programs that use NS-ARPA that you expect to run concurrently on the system. Enter the maximum number of user programs plus the number of NS-ARPA subsystem programs for the local node. The default *`number of user programs`* is 8, so the default number of active NS-ARPA programs is 14 if you have NFT, 13 if you do not.

Internal Resources: Every active NS-ARPA program requires a NetIPC root socket, system resource number, and NetIPC user record. The NetIPC user record is not released until the program terminates; it is not released when the program terminates its datacomm activity. The number of NetIPC user records allocated is the configured number of maximum active NS-ARPA programs plus 10. NetIPC uses the 10 additional user records for resource management.

If you configure the minimum value (1), you will only be able to run one NS-ARPA program at a time at the local node.

```
* Enter the maximum number of active NS sockets [9..(nnn)..200]:
```

The maximum number of active NS-ARPA sockets is the maximum number of concurrently active NetIPC root, call, and VC sockets for the NS-ARPA subsystems and user programs.

The default maximum number of active NS-ARPA sockets is determined by the following formula:

```
number of NS-ARPA subsystem sockets + 3 * number of user programs
```

The *number of NS-ARPA subsystem sockets* is 10 if you do not have NFT, 19 if you do. The *number of user programs* is multiplied by three because each program requires a root socket and a call socket in addition to any VC sockets. The default *number of user programs* is 8, so the default maximum number of active NS-ARPA sockets is 37 if you do not have NFT, and 46 if you do. Enter the maximum number of NS-ARPA sockets that you expect to be concurrently active, including NS-ARPA subsystem sockets.

If you configure the minimum value (9), you will only be able to run one NS-ARPA program with one VC connection at a time at this node. You must not activate NFT, RPM, FTP, and TELNET services.

Internal Resources: The maximum number of active NS-ARPA sockets is the number of NetIPC socket records and TCP path records allocated at this node.

Nodal Registry

The Nodal Registry manages the Nodal Path Reports that the transport and services use to establish connections with remote nodes. For more information, refer to the section "Network Configuration Planning," subsection "Nodal Registry and Probe Configuration."

The local node is automatically put into the Nodal Registry by NSINIT. Refer to the "Network Management" subsection earlier in this section.

```
*++            Nodal Registry           ++*
*
*       Maximum number of Connect-Site path reports is:  nnn.
*          (The above value is derived from previous responses.)
*       Maximum number of Nodal path reports is:              nnn.
*       Maximum number of Name records is:                    nnn.
* Do you want to modify these values [Y/(N)]?
```

Enter `Y` (YES) to modify Nodal Registry resources; otherwise, enter `N` (NO).

If you enter `N`, (NO), NSINIT skips the next two questions. Otherwise, NSINIT prompts:

```
* Enter the maximum number of Connect-Site path reports [1..(nnn)..1000]:
```

A connect-site path report is created and stored in DSAM each time a program calls IPCLookUp. The DSAM storage space is not released until the program calls IPCShutDown for the path descriptor. Enter the maximum number of concurrently active path reports that you expect to have.

If you configure the minimum value (1), you will only be able to run one NS-ARPA program at a time at this node.

Internal Resources: NSINIT reserves space in DSAM for connect-site path report storage according to the configured maximum number of connect-site path reports. When this storage is exhausted, NetIPC may fail IPCLookUp requests.

The default is the `maximum number of active NS-ARPA sockets` from the previous question.

```
* Enter the maximum number of Nodal path reports [2..(20)..1000]
```

Nodal Path Reports contain address, transport protocol, and NS Common Services and ARPA Services information. The local node requires a Nodal Path Report for every node in the internet that supports NS Common Services and ARPA Services which the local node will communicate with, plus one for the local node. Enter the maximum number nodes in the network that will support NS Common Services and ARPA Services.

Internal Resources: NSINIT reserves space in DSAM for Nodal Path Reports according to the maximum number of Nodal Path Reports. When this storage is exhausted, no new Nodal Path Reports may be added to the Nodal Registry (via NRINIT or as a result of Probe queries).

```
* Enter the maximum number of name records (the default is derived
* from previous responses) [1..(nnn)..nnn]
```

NetIPC creates a name record when a user names a socket with an `IPCName` call or gives a socket away with an `IPCGive` call.

The default maximum number of name records is half the maximum number of active NS-ARPA sockets, and the upper bound is three times the maximum number of active NS-ARPA sockets. Enter the number of name records for the local node.

Internal Resources: Name records reside in DSAM. The number of name records allocated is the configured maximum plus 2. NetIPC uses the 2 additional name records for resource management.

TCP

```
 *++   Transmission Control Protocol (TCP)   ++*
*
* Initial segment size in Bytes:                          4096
* Retransmission backoff algorithm:                       Exponential
* Initial retransmission timeout value in seconds:        5
* Retransmission Smoothing parameter Alpha:               9
* Retransmission Smoothing parameter Beta:                20
* Do you want to modify these values [Y/(N)]?
```

Enter `Y` (YES) to modify Nodal Resources; otherwise, enter `N` (NO).

Note HP recommends that you do not modify TCP options.

If you enter `N` (NO), NSINIT skips to “IP.” Otherwise, NSINIT prompts:

```
* Enter the initial segment size [1024..(4096)..8000]:
```

TCP accepts arbitrarily long sequences of user data and divides them into segments for transmission. If the IP network segment size for a given connection is less than the TCP segment size, IP will request TCP to use a smaller segment size for the connection. For the best performance over LAN links, enter a multiple of 1450 (such as 4350). For the best performance over other links, enter the largest buffer size. Enter the initial segment size in bytes.

Retransmission

The next three questions are used for retransmission. TCP has a timer to force retransmission of unacknowledged segments. The retransmission timeout value is the number of centiseconds that the timer will wait before TCP retransmits unacknowledged segments. (TCP dynamically updates this value, as described in the following subsection.) If TCP receives a valid acknowledgement for all unacknowledged segments, it clears and restarts the timer. If the timer goes off, TCP retransmits all unacknowledged segments, and increases the amount of time that it waits before retransmission (also called doing a backoff on the retransmission timeout value).

Retransmission Timer Update. Each time TCP receives a valid acknowledgement for a previously unacknowledged segment, it determines the new retransmission timeout value. TCP first determines the current round-trip time (`rtt`). The `rtt` is the number of centiseconds that have elapsed since the oldest segment acknowledged by the current acknowledgement was transmitted. TCP then calculates the new *smoothed round-trip time* (`srtt`) with the following formula:

```
new srtt = (alpha * old srtt) + ((1 - alpha) * rtt)
```

TCP then calculates the retransmission timeout value with the following formula:

```
retransmission timeout value = beta * srtt
```

Retransmission Back-Off. If the retransmission timer goes off, TCP retransmits all unacknowledged segments, and does a back off on the retransmission timeout value by multiplying the timeout value by a back-off multiplier. TCP determines the back-off multiplier according to the number of times that a segment has been retransmitted. TCP uses the following back-off multipliers:

Number of	Linear	Exponential
Retransmissions	Back-off Multiplier	Back-off Multiplier
1	1.0	1
2	1.5	2
3	1.7	4
4	2.5	8
5	3.0	16
6	3.5	32
7	5.0	64
8	8.0	128

In addition, the `retransmission timeout value` times the `Back-off Multiplier` must be between 1 and 30 seconds.

```
* Do you want Exponential (E) or Linear (L) backoff [(E)/L]?
```

Enter `E` for an Exponential backoff algorithm, `L` for Linear.

```
* Enter the value for Alpha [1..(9)..10]:
```

Enter a value for Alpha in tenths of seconds.

```
* Enter the value for Beta [1..(20)..20]:
```

Enter a value for Beta in tenths of seconds.

IP (DCN, GT)

```
*++ DCN  ++*
* Enter information for each Directly Connected Network (DCN).
* Format:
*
* <local IP addr>, [<subnet mask>,]RTR, <segsize: [1200..(8000)]>
* <local IP addr>, [<subnet mask>,]802/LAN/ETHERNET, <segsize: [1200..(1514)]>,
*                  <link LU>, (E)/NE, <station addr>
*
* Where E = Enable, NE = do Not Enable. Type /E to end.
*
*DCN:
```

You must enter DCN information if the local node has ROUTER/1000 (RTR) or LAN (802, ETHERNET, or both LAN) LI types. IP uses the DCN information to determine which Router/1000 or LAN networks the local node belongs to and to determine the local IP addresses for the node. For more information on the DCN, refer to the section “Network Configuration Planning,” subsection “IP Directly Connected Networks (DCN).”

Make an entry for each Router/1000 or LAN network to which the local node is directly connected. You cannot use more than three LAN link interface cards for NS-ARPA/1000 per system. Each entry contains the following fields:

The `local IP addr` is the IP address assigned to the local node for that network. For more information on IP addresses, refer to the section “Network Configuration Planning,” subsection “IP Addresses.”

If the node is part of a subnetwork (subnetting is used), enter the `subnet mask` for each local IP address. If subnetting is not used, skip this field. If you are skipping the subnet mask field, do not put a comma placeholder. The subnet mask specifies the portion of the IP address that is the subnet number. Refer to the subsections “IP Addresses with Subnetting” and “Subnet Masks” in Section 3, “Network Configuration Planning.”

Enter `RTR` if the local node is connected to that network via a Router/1000 Link Interface (LI).

Enter `802` if it is connected via an IEEE 802.3 (802) LI.

Enter `ETHERNET` if it is connected via an Ethernet LI.

Enter `LAN` for both IEEE 802.3 and Ethernet LI. HP recommends `LAN`.

In the `segsize` field, enter the network segment size. The network segment size is the largest size buffer (in bytes) that the link level protocol will accept; IP uses this value to determine if it needs to segment a message before sending it to the link level. For the default value, leave this field blank, but enter a comma placeholder if you enter any subsequent fields.

HP recommends that you use the default network segment size. In most cases, it is more efficient to transmit a few large messages than several small messages. Additionally, if an intervening network uses a smaller network segment size, IP will further segment any messages to accommodate the smaller network segment size, but will not reassemble the messages until they arrive at the final destination.

However, you may want to use a smaller network segment size if the line is unreliable or noisy. Less data may be retransmitted if you use a smaller segment size and overall throughput may improve.

Note that you must have enough SAM to receive messages of the size that you configure for `segsize`.

If the network is connected via a Router/1000 (RTR) LI, you do not have to enter any more fields. NSINIT will prompt you for more Router/1000 information later.

If the network is connected via an IEEE 802.3, Ethernet, or both (LAN) LI, enter information for the following three fields. In the `link LU` field, enter the link LU. This is the LU of the LAN interface card for the network indicated by the specified local IP address.

Enter `E` in the next field if you want NSINIT to enable the link at initialization time; enter `NE` if you do not want NSINIT to enable the link.

In the last field, enter a user-assigned LAN station address if you want to override the LAN card’s address (the address in NOVRAM). *HP recommends that you use the HP-assigned station address*. To use the default station address, skip this last field.

When you have entered the last node, enter `/E` to end data entry. For example:

```
DCN:  192.006.001.128,RTR
DCN:   192.006.250.128,802,76,E,08-00-09-1C-77-1A
DCN:  192.006.250.123,802,82,E
DCN:  /E
```

Internal Resources: IP uses the DCN entries to build the Local IP Address List and to add entries for the local node in the Gateway Table.

As noted above, the value of *`segsize`* affects the amount of SAM you must have to receive messages from the link.

At revision 5240, the gateway half dialogue has been removed from NSINIT. A second `/E` is required following the DCN section. NSINIT puts in the second `/E` if building an output file. This was done to be compatible with previous NSINIT answer files:

```
** Internal Use Only **
/E
*++ GT ++*
* Enter information for each Gateway (GT).
* Format:
*
* <Dest IP Net>, <Gateway IP>, <Hops: [1..(100)]>
*
* Where:   Dest IP Net = IP address of any node on a remote net
*          Gateway IP  = DCN IP address of Gateway to use
*          Hops        = Maximum allowed IP hops to Dest IP Net
* Type /E to end.
*
*GT:
```

Each entry defines a route to a specified remote network. These entries form the Gateway Table (GT). Each entry contains the following fields:

The *`Dest IP Net`* is the IP address of any node on the remote network. It can be the address of any node because IP uses only the network portion of the address. If the remote node has more than one IP address, this must be the IP address associated with the remote network for which you are making an entry.

You can specify *`000.000.000.000`* for the *`Dest IP Net`* address to indicate that the associated *`Gateway IP`* (defined below) will be the default gateway. The default gateway is used to route any message for a remote network not defined in the Gateway Table. Refer also to the subsection “IP Gateway Table (GT)” in Section 3, “Network Configuration Planning.”

The *`Gateway IP`* is the IP address of the node through which all messages for the specified remote network are sent. The network address portion of *`Gateway IP`* must match the network address portion of an entry in the DCN section. The gateway node must be on the same network as the local node. If the *`Dest IP Net`* address is *`000.000.000.000`*, the *`Gateway IP`* is the default gateway.

The `Hops` is the maximum number of gateways to route through before reaching the destination network. The valid range is 1 to 100; default is 100. *HP recommends that you use the default value*.

If IP routes a message through more networks than the value for `Hops` without reaching the destination, it drops the message. Therefore, `Hops` should be the greater than or equal to the total number of networks in the internetwork.

When you have entered the last node, enter `/E` to end data entry.

For example:

```
GT:   192.006.001.001,192.006.250.007,2
GT:   192.006.251.003,192.006.250.008,3
GT:  /E
```

Note NS-ARPA/1000 can communicate with machines not directly connected on the LAN (i.e., over X.25 or point-to-point links) if these links are connected to the LAN through gateways. Specify the appropriate *<Dest IP Net>*, *<Gateway IP>*, *<Hops>* answers to the Gateway Table question to reach these remote machines.

```
* Enter the maximum number of path records for IP. The default is derived
* from previous responses [6..(nnn)..200]:
```

IP path records are used to establish routes between nodes for NS-ARPA/1000 common transactions (except DS/1000-IV transactions over Router/1000, Bisync, or X.25 links). You must have an IP path record for each concurrent, unique IP connection, or each concurrent connection that has a unique triplet of source node, destination node, and upper-level protocol (TCP or PXP), including IP connections for store-and-forward traffic through the local node. Because IP path records are only needed for unique IP connections, sockets can share IP path records. For example, two sockets at the same source node with a TCP connection to sockets at the same destination node can use the same IP path record.

Path records used for TCP are not released until all TCP connections using it are released. If path records used for PXP are idle, they may be reused if there are no free IP path records, and are deallocated if they are idle for 5 minutes. If IP path records used for store-and-forward traffic are idle, they may be reused if there are no free IP path records, and are deallocated if they are idle for 5 minutes.

The default is the configured maximum number of NS-ARPA sockets plus 2 for store-and-forward traffic. Enter the maximum number of IP path records. IP will reserve 4 path records for store-and-forward traffic.

Internal Resources: IP path records reside in DSAM.

Address Resolution

NSINIT asks the following questions only if you have an 802, Ethernet, or LAN LI in the local node. If you do not have a 802 or LAN LI, skip to the subsection “DS/1000-IV Compatible Links.” The Probe protocol allows IEEE 802.3 LAN nodes to query one another for Nodal Path Reports and other address information. For more information, refer to the section “Network Configuration Planning,” subsection “Nodal Registry and Probe Configuration.” (Also, see the example on page B-15.)

```
*+++            ADDRESS RESOLUTION        +++*
* Maximum number of active PCB Records:           10
* Retry interval timeout in centi-seconds:        100
* Proxy Nodal Registry Server:                    No
*
* Do you want to modify these values [Y/(N)]?
```

Enter Y (YES) to modify ~~Probe~~ defaults; otherwise, enter N (NO). If you enter N (NO), NSINIT skips the next three questions. Otherwise, NSINIT prompts:

```
* Enter the maximum number of PCB records. The maximum is based on previous
* responses. [1..(10)..nn]:
```

Each active Probe query, for a Nodal Path Report, LAN station address, requires a Probe Control Block (PCB).

The maximum number of active PCB records is the configured maximum number of NS-ARPA programs. Enter the maximum number of concurrently active Probe Control Blocks.

```
* Enter the retry timeout interval in centi-seconds [50 .. (100) .. 200]:
```

The Retry Timeout Interval is the number of hundredths of seconds that Probe will wait for a reply before timing out. Probe will retry up to 3 timeout intervals before returning an error.

Enter the Retry Timeout Interval in tenths of seconds.

The following questions are asked only if you have an 802 or LAN LI.

```
* Do you want to be a Proxy Nodal Registry server [Y/(N)]?
```

If you enter Y (YES), the local node will respond to requests for Nodal Path Reports on behalf of other nodes, including nodes that are not on the LAN, such as Router/1000 nodes.

If you have a Nodal Registry Server for each LAN, you need to configure Nodal Path Reports only at the Nodal Registry Server. Other nodes on the LAN can then query the Nodal Registry Server for Nodal Path Reports.

```
*++ Default IEEE-802 Multicast Addresses for Probe:
* Target Address:         09-00-09-00-00-01
* Proxy Address:          09-00-09-00-00-02
*
* Do you want to modify these values [Y/(N)]?
```

Enter `Y` (YES) if you want to change the multicast address for any node on the LAN. However, *HP recommends that you do not change the default multicast addresses*. If you do, you must configure the same multicast addresses at all nodes in the network.

NSINIT asks the remaining Probe questions if you enter `Y`. Otherwise, NSINIT skips to "DS/1000-IV Compatible Links."

```
* For each IEEE-802 LU enter the following:
* Target multicast address:  HH-HH-HH-HH-HH-HH
* Proxy multicast address:   HH-HH-HH-HH-HH-HH
* Where H is one hexadecimal digit.  Type /E to end.
```

NSINIT prints the following prompt for each IEEE-802.3 transmit LU in the local node:

```
* IEEE-802 LU nnn:
* Enter a target address [/D = 09-00-09-00-00-01]:
```

Enter the Target Address for the LU *nnn*. The Target Address is used for inbound and outbound Name and Virtual Address Requests.

If you have LAN/1000 Node Manager software and the appropriate multicast address file for your card (the file `/files802/addr`*hhhhhhhhhhhh*`/mcast.txt`, where *hhhhhhhhhhhh* is the card's station address) exists on your system with data in it, *the multicast addresses for NS-ARPA must be included in the multicast address file*.

For more information, refer to Section 3, "Network Configuration Planning."

```
* Enter a proxy address [/D = 09-00-09-00-00-02]:
```

Enter the Proxy Address for the LU *nnn*. The Proxy Address is used for Proxy Name Requests.

If you have LAN/1000 Node Manager software and the appropriate multicast address file for your card (the file `/files802/addr`*nnnnnnnnnnnn*`/mcast.txt`, where *nnnnnnnnnnnn* is the card's station address) on your system, *the multicast addresses for NS-ARPA must be included in the multicast address file*.

For more information, refer to Section 3, "Network Configuration Planning."

Router/1000

NSINIT asks the following questions if you responded `Y` to `Do you want any DS/1000-IV Services?` or if you entered a RTR link in the DCN table. Otherwise, NSINIT skips to "Security Codes."

```
*++         Router/1000           ++*
*
* Enter the Router/1000 address of the local node.[1..32767]:
```

This must be an integer between 1 and 32767, inclusive. Address 0 (zero) is not allowed.

```
* Enter the MA retry limit [1..(4)..13]:
```

Enter the maximum number of times that you want Message Accounting to retry (retransmit) an unacknowledged message. This value is only used if you specify Message Accounting (`MA`) for a node in the Nodal Routing Vector (NRV), below. The product of the MA retry limit and the MA timeout value (configured in the NRV) must be less than or equal to the Master timeout value (configured in the DS/1000-IV Compatible Services section of the dialogue).

```
* Enter the Nodal Routing Vector (NRV) information. Type /E to end. Format:
* <Node Number>, <IP addr>, <LU>, <LU T/O>, <Level>, N, MA, <MA T/O>
*
* Where:
*    node number  : Router/1000 node address
*    IP Address   : IP Address. Format: n.n.n.n where n = 1..255
*    LU           : Link LU if rerouting is not used to access the node
*    LU T/O       : Link timeout in seconds
*    level        : DS/1000 software level
*    N            : Neighbor bit, set if this node is a neighbor to local node
*    MA           : Use Message Accounting in transactions to this node
*    MA T/O       : Message Accounting time out in seconds
*
*NRV:
```

NRV entries are used to set line characteristics for lines to remote HP 1000s and to tie IP addresses to Router/1000 addresses. Enter an NRV entry for every node on the local network connected via a Router/1000 link and for every remote HP 1000 *in the internet* that supports DS/1000-IV Compatible Services, including DS/1000-IV (91750) nodes. For more information on the NRV, refer to the section "Network Configuration Planning," subsection "Router/1000 Nodal Routing Vector."

To get the default value for any field, leave it blank. However, if you leave a field blank but include any field that follows it, you must enter a comma as a placeholder.

The *`node number`* is the Router/1000 address. This must be an integer between 1 and 32767, inclusive. Address 0 (zero) is not allowed.

The *`IP Address`* is the remote node's IP address.

If the remote node is an NS-ARPA/1000 node and is a member of a Router/1000 network, *this must be the IP address associated with the Router/1000 network*.

If the remote node is not a member of a Router/1000 network and has more than one IP address, you can enter any one of its IP addresses.

If you have DS/1000-IV (91750) nodes and an internetwork (you have more than one network), you must assign NS-ARPA/1000 *guardian* (91790) nodes. Specify the guardian node by entering the IP address of the guardian node in the Nodal Routing Vector (NRV) entry for the DS/1000-IV node.

Do not assign a guardian node if the local NS-ARPA/1000 node and the DS/1000-IV (91750) node are members of the same DCN. For more information on guardian nodes, refer to "Guardian Nodes for DS/1000-IV (91750) Nodes" in the "Network Configuration" section.

The *`LU`* field is used for the transmit LU of the transmit/receive LU pair of the Router/1000 link interface card to the remote node. Enter the transmit LU here only if the remote node and the local node are members of the same Router/1000 network and NS-ARPA is *not* to use Dynamic Rerouting to the remote node.

Leave this field blank if: the remote node and the local node are not members of the same Router/1000 network; or if NS-ARPA is to use Dynamic Rerouting to the remote node. HP recommends that you use Dynamic Rerouting over HDLC RTR links.

The *`LU T/O`* is the transaction timeout in seconds. This is the maximum number of seconds the local system will wait for a reply from the remote system before reporting a timeout error. If you leave this field blank, NSINIT will use the Master timeout value (configured in the DS/1000-IV Compatible Services section of the dialogue).

You should have the same link timeout for the local and remote system.

Default: the Master timeout value (configured in the DS/1000-IV Compatible Services section of the dialogue).

The *`Level`* is the software level. Enter `0` for DS/1000 (91740) nodes; enter `1` for DS/1000-IV (91750) and NS-ARPA/1000 (91790) nodes. If you enter `0`, NSINIT will attempt to schedule the 91740 message converters, INCNV and OTCNV, at initialization time.

Default: 1.

Enter *`N`* in the fifth field if the node is a neighbor (there is a direct link to this node). If Dynamic Rerouting is used (you did not specify an LU for this node), this field is optional.

Default: The remote node is not a neighbor.

Enter *`MA`* in the sixth field if NS-ARPA is to use Message Accounting for transactions with this node. If you enter *`MA`* for one or more nodes, NSINIT will initialize Message Accounting (MA) and will attempt to schedule the MA timing module, MATIC.

Default: Do not use MA.

The *`MA T/O`* is the Message Accounting timeout value for transactions with this node. This is the number of seconds that Message Accounting will wait for an acknowledgment to arrive for a message before timing out. MA will then retransmit (retry) the message. MA will retransmit a message as many times as the MA Retry count. The MA timeout value times the MA retry limit (configured in the previous question) must be less than or equal to the Master timeout value (configured in the DS/1000-IV Compatible Services section of the dialogue).

Default: 3.

Internal Resources: Each node specified is stored in the NRV which resides in DSAM.

Router/1000 Links

NSINIT asks the following questions if you responded `Y` to `Do you want any DS/1000-IV Services?` or if you entered a RTR link in the DCN table. Otherwise, NSINIT skips to "DS/1000-IV Compatible Links (RTE-MPE)."

```
* Enter the LU information for Router/1000.  Type /E to end.
* Format: <Link LU>, (E)/NE, (RR)/NR, <cost: [(1)..99]>
*
* Where:  E = Enable, NE = do Not Enable
*         RR = Use ReRouting,  NR = do Not use ReRouting
*
*Link:
```

Make an entry for each Router/1000 link in the local node. If the local node has no Router/1000 links, enter `/E`. If you do enter one or more Router/1000 links, NSINIT will attempt to schedule GRPM at initialization time.

To get the default value for any field, leave it blank. However, if you leave a field blank but include any field that follows it, you must enter a comma as a placeholder.

The *`Link LU`* is the transmit LU of the transmit/receive LU pair for the link interface card to the remote node.

In the next field, enter *`E`* to enable the LU at initialization time; enter *`NE`* otherwise.

In the next field, enter *`RR`* if the link is an HDLC RTR link and you want Dynamic Rerouting over this link, *`NR`* if you do not. HP recommends that you use Dynamic Rerouting if the link is an HDLC RTR link.

You can only specify *`NR`* if you already specified the LU in the NRV. If you have already specified the LU in the NRV and then specify *`RR`*, NS-ARPA will override the specification in the NRV and enable Dynamic Rerouting for the link.

In the last field, enter a cost value for the link if it is a rerouting link. The cost value is used to determine routes between nodes. Dynamic Rerouting will select the best route between nodes according to the lowest cost value. Assign cost values relative to various link speeds or connection costs for telephone connections. For example, if you have a hardwired link with a 9600 baud line speed, you could assign it a cost value of 1, and if you have a modem link with a 300 baud speed, you could assign that link a cost value of 99. You can also assign costs to off-load traffic over links.

Internal Resources: Each Dynamic Rerouting link is stored in the Link Vector (LV). Each link without Dynamic Rerouting is stored in the Non-Rerouting Table. NSINIT also uses the Router/1000 LU information to build the Cost Matrix; the number of entries is the number of Link Vector entries times the number of entries in the NRV. The LV, Non-Rerouting Table, and Cost Matrix reside in DSAM.

DS/1000-IV Compatible Links (RTE-MPE)

NSINIT asks the following questions only if you entered `Y` to `Do you want DS/1000-IV 1000-3000 services`. Otherwise, NSINIT skips to "Security Codes."

```
* Enter the maximum number of concurrent X.25 pool LUs used for DS/1000-IV
* to DS/3000 communication [(0)..255]:
```

The maximum number of concurrent X.25 pool LUs determines how many entries will be reserved in the DS/1000-IV-DS/3000 LU table for X.25 pool LUs. Enter the maximum number of X.25 connections for DS/1000-IV Compatible Services to HP 3000s that you expect to be concurrently active at any time.

Internal Resources: DS/1000-IV-DS/3000 LU table entries reside in the SMB.

Next, NSINIT prompts you for each Bisync link used for DS/1000-IV Compatible Services (1000-3000), one at a time:

```
* Enter a PSI-Bisync LU. Type /E to end:
```

The entries for this question are used to build the DS/1000-IV-DS/3000 LU table. Enter the transmit LU of the transmit/receive LU pair for the link interface card for each PSI Bisync link used for DS/1000-IV Compatible Services (1000-3000). Enter `/E` when you have entered all PSI Bisync LUs used for DS/1000-IV Compatible Services.

Internal Resources: DS/1000-IV-DS/3000 LU table entries reside in SMB.

If you did not enter any Bisync LUs, NSINIT skips to "Security Codes." Otherwise, NSINIT prompts:

```
*++      BISYNC ID SEQUENCES:    ++*
*      Local ID sequence  : none
*      Remote ID sequence : none
* Do you want to modify these values [Y/(N)] ?
```

If you have Bisync dial-up links used for DS/1000-IV Compatible Services, you can assign ID sequences to limit access to the HP 1000. For more information, refer to "Names and Addresses" in section 2.

Enter `Y` if you want to change the local or remote ID sequence. If you do not have any Bisync dial-up links for DS/1000-IV Compatible Services, enter `N`.

The next two questions are only asked if you enter `Y` to the previous question.

```
* Enter a local ID sequence [/D = none, Type /E to end] :
```

Enter a local ID sequence of 1 to 15 characters.

```
* Enter a remote ID sequence [/D = none, Type /E to end] :
```

Enter a remote ID sequence of 1 to 15 characters.

Internal Resources: The ID sequences reside in the system labeled common area.

Security Codes

```
* Enter the Network security code for this node [1..2 char]:
```

The Network security code is required by NSINIT to shut down NS-ARPA; by Remote I/O Mapping to establish Remote I/O maps; and by DSMOD to modify timing parameters and to change the NRV.

Enter two alphanumeric characters. Be sure to remember the security code; you cannot retrieve it later.

```
* Enter the Network User’s security code for this node [1..2 char]:
```

Users must enter this security code to execute the REMAT `SW` (SWITCH) command; thus it is used to restrict REMAT access to remote nodes. Enter two alphanumeric characters; the first character must be alphabetic. Be sure to remember the security code; you cannot retrieve it later.

Protocol EMAs

If the NSINIT input device is interactive or if an output device or file is specified, NSINIT displays or prints EMA information. NSINIT displays the number of EMA pages available and the number of EMA pages used by NSINIT. The number of EMA pages available is set when NSINIT is linked.

```
*++   EMA Usage   ++*
*  3 pages available   1 pages used
```

Required Resources

If the NSINIT input device is interactive or if an output device or file is specified, NSINIT prints a summary of the resources required for NS-ARPA according to your previous dialogue responses:

```
*++ System resources required for this file ++*
* DSAM table size in words:                 nnnnnnnn
* Number of RTE Class Numbers:              nnn
* Number of RTE Resource Numbers:           nnn
* SMB size in words:                        nnn
```

You can use these values to help you allocate system resources, especially if you have to regenerate or reboot the system.

The *DSAM table size* is the number of words required in DSAM for NS-ARPA data structures, including tables such as the IP GT, and data structures such as NetIPC Socket Records. This number does *not* include DSAM for message buffers.

The *Number of RTE Class Numbers* is the number of RTE class numbers required for NS-ARPA monitors, including all monitors specified for scheduling, whether or not the monitor is present and restored (RPed). This number does not include user application requirements, or the two class numbers required for LAN Node Manager. If you did not specify that you want NSINIT to schedule NSTRC, this number does not include the two class numbers required for NSTRC.

The *Number of RTE Resource Numbers* is the number of RTE resource numbers required for NS-ARPA monitors. This number does not include the resource numbers required for NetIPC processes.

The *SMB size* is the number of words required for the SMB. SMB is used for the tables of some DS/1000-IV Compatible Services.

Start Up Dialogue

If you entered 1 (build output file) for the NSINIT operation option, NSINIT skips the remaining questions and terminates. If you entered 3 (initialize NS-ARPA) for the NSINIT operation option, NSINIT skips the remaining questions and initializes NS-ARPA.

If you entered 2 (build output file and initialize NS-ARPA) for the NSINIT operation option, NSINIT asks the following question before initializing NS-ARPA:

***Do you want to start up the NS network on this system at this time [(Y)/N] ?**

Enter Y to initialize NS-ARPA, otherwise enter N. If you enter N, NSINIT will terminate.

If you enter Y, NSINIT prints the number of pages required for the DSAM SHEMA (#DSAM) for the NS-ARPA tables. If you specified NFT and NSINIT was able to relocate the NFT message file to DSAM, this value will include the 3500 words required for the NFT error message file.

**** (138) NSINIT: Min. SHEMA size needed is buffer space + table space of :*nn* ****

This is an informational message only and not an error.

NSINIT will attempt to schedule the monitors required for the features and services configured or requested in the NSINIT dialogue. Refer to the section “Internal Resources” for information on the monitors required.

Shut Down Dialogue

You must have superuser capability to shut down NS-ARPA. If you entered 4 (shut down NS-ARPA) for the NSINIT operation option, NSINIT prints the following statistics about the local node:

```
*+++ Current Active Statistics: .+++*
* Transaction Control Blocks (TCB):      nnnn
* Process Number List Entries (PNL):     nnnn
* Remote Session Table (#POOL):          nnnn
* Call Sockets:                          nnnn
* VC Sockets:                            nnnn
```

Transaction Control Blocks (TCB) is the number of active TCBs or DS/1000-IV transactions. A program or remote access that uses DS/1000-IV Compatible Services (such as REMAT, RMOTE, DLGON, HELLO, Remote I/O Mapping) is a DS/1000-IV transaction. A non-zero value indicates active DS/1000-IV Compatible Service transactions.

Process Number List Entries (PNL) is the number of active PNL entries, which are TCBs used for transactions to HP 3000s. A non-zero value indicates active DS/1000-IV Compatible Service (RTE-MPE) transactions.

Remote Session Table (#POOL) is the number of active #POOL entries, or remote DS/1000-IV Compatible Service programs (including monitors, such as REMAT) accessing the local node. A non-zero value indicates DS/1000-IV Compatible Service (RTE-RTE) program is currently accessing the local node.

Call Sockets is the number of active Call Sockets. If this value is one and NFTMN is present, it indicates that only the NFT call socket is active; there are no active user call sockets.

VC Sockets is the number of active VC (Virtual Connection) Sockets. A non-zero value indicates active user VC sockets. You should exercise caution in shutting down the network when this value is not zero.

NSINIT asks the following question before shutting down NS-ARPA:

```
*Do you still want to shut down NS on this node [(Y)/N] ?
```

Enter Y to shut down NS-ARPA, N otherwise.

When NS-ARPA prepares to shut down, it does not wait for any NS-ARPA transactions, NFT transaction, NetIPC programs, FTP, or TELNET sessions to complete. If you shut down NS-ARPA while any of these are active, the programs may be terminated abnormally.

If you enter Y (YES), NSINIT prompts you for the Network Security Code:

```
* Enter the Network security code for this node [1..2 char]:
```

Enter the Network Security Code for the local node. If it is correct, NSINIT will shut down the node.

LAN Links

NSINIT shutdown does not disable LAN links. The LAN interface card will continue to accept NS-ARPA packets, but the packets will be undeliverable (*orphan packets*). For more information on orphan packets, refer to *HP 12076A LAN/1000 Link Installation Manual* or *HP 12076A LAN/1000 Link Node Manager's Manual*.

Removing NS-ARPA/1000 Programs

Remove NS-ARPA/1000 programs using RTE `OF` commands. Refer to the command file, `/NS1000/CMD/OF_NS.CMD` as an example and change it according to your system. You must shut down NS-ARPA with NSINIT before running such a command file.

Editing NSINIT Answer Files

As noted earlier, NSINIT can produce answer files. The following subsections describe how to edit NSINIT answer files.

Basic Editing

Any lines with an asterisk (`*`) in column one are comments. The prompts are placed in the file as comments. If you edit the file, you do not need to change the comments. However, you may wish to do so to ensure the readability of the file.

The Troubleshooting section of *NS-ARPA/1000 Maintenance and Principles of Operation* has information on debugging errors that may occur in NSINIT files.

Possible Answer File Changes

This subsection lists changes that you can easily make to an existing NSINIT answer file with EDIT/1000. With a lot of effort, you can modify any file, but the changes described here do not require you to add responses not already in the answer file. The easily made changes are:

- In general, you can change numeric values in existing responses (buffer sizes, time out values, table sizes, number of programs, IP addresses, or LUs). Note that some values affect the default values for later responses.

- You can also add or delete entries in the following tables, as long as there was at least one entry in the table to begin with, and you do not delete all the entries in any table:

  ```
  GT (IP Gateway Table)
  NRV (Router/1000 Nodal Routing Vector)
  LINK (Router/1000 Link LU Table)
  ```

Inadvisable Answer File Changes

Changing responses to some prompts cause the NSINIT dialogue to change. HP recommends that you do not modify these responses with EDIT/1000 because it is difficult to complete the changes without making a mistake.

Unless you are sure that the response you are attempting to change with EDIT/1000 will not affect the NSINIT dialogue, you should re-run NSINIT and build the output file from scratch.

Making EDIT/1000 changes to the following areas can cause problems:

- Changing link types: adding or deleting LI types in the DCN section changes the network boundaries of the node being configured.
- Changing the `Y/N` response to a set of options (`Do you want to modify these values?`).

10

INETD—Internet Network Services Daemon

Overview

The Internet Network Services Daemon, INETD, is a monitor that listens for incoming connections and schedules the appropriate server program to handle the connection. In NS-ARPA/1000 systems, INETD handles incoming connections for the FTP, TELNET, SMTP (Mail/1000), Remote Print Spooling, and Time Server Protocol services supplied by HP. INETD can also schedule user-written NetIPC server programs. INETD must be running before other hosts can connect to these services on the local host. INETD must be RPed prior to running NSINIT.

INETD is also able to control which remote systems can access the network services that it monitors. For each service, system managers can specify the IP addresses of remote systems that should be denied or allowed access to that service. In addition, a record of connections can be logged in the INETD log file.

INETD uses three disk files. The configuration file (`/etc/inetd.conf`) and services mapping file (`/etc/services`) are read by INETD at startup. After startup, INETD uses the log file (`/etc/inetd.log`) to record error messages and the optional connection information. Users can specify a configuration file name or a log file name when INETD is started.

INETD is scheduled by NSINIT with no options. Thus, the default configuration file, `/etc/inetd.conf`, and log file, `/etc/inetd.log`, are used. INETD can be restarted to change these files or to re-read the configuration file. The following section lists the INETD runstring options.

INETD provides the time service (Time Server Protocol) using internal routines. The time service returns the number of seconds since midnight, January 1, 1900 in a 32-bit (4-byte) unsigned integer. If the system time zone was set via the SYSTZ program, the time is converted to Coordinated Universal Time (UTC). This service is provided only over TCP and is compatible with the RDATE client program. An entry in the `inetd.conf` file with the service specified as “time” or “37” and server “internal” is required to enable the time service.

INETD Runstring

To start up a primary INETD if no INETD is running, use the following syntax. If this form is used when there is an INETD already running, the error message, `"An inetd is already running"`, is displayed and the command has no effect.

Syntax

```
xq inetd [-l logfile]
```

Parameters

logfile The path name of the file to log connection to services, along with information about errors encountered. The default file name is `/etc/inetd.log.`

To reconfigure INETD the syntax is as follows. In this form, if an INETD is already running, this command shuts it down and starts up a new one. If no INETD is running, the INETD scheduled by this command becomes the primary INETD.

Syntax

```
xq inetd -c [configfile]
```

Parameters

configfile The path name of the file containing the INETD configuration information. The default configuration file is `/etc/inetd.conf` or `/system/inetd.conf.`

To terminate INETD, enter the following command. If an INETD is running, this command shuts it down.

Syntax

```
inetd -k
```

Discussion

Note that only one option may be specified in the runstring. Only superusers may shut down an INETD that is running.

INETD Configuration

INETD reads the file `/etc/inetd.conf` or `/system/inetd.conf` at startup to determine for which services to listen. If this file is changed, INETD needs to be shut down and restarted with `inetd -c`. This file may contain three types of lines:

- a comment line with an asterisk (*) or pound (#) in column 1
- a service definition beginning in column 1
- security/logging restrictions starting in column 2 or greater.

Service Definitions

Service definitions start in column 1 of the line, and are in the following format:

Syntax

```
    service sockettype protocol (no)wait user server [-flags] runstring
```

Parameters

service
The name of a service defined in the `/etc/services` file (described in the "Services Mapping" subsection) or a TCP port address.

Ports 1-255 are reserved by the Internet Network Information Center for registered "well-known" services (such as SMTP).

sockettype
Must be `stream` to correspond to the connection-based TCP protocol. It is mandatory to enter this field for compatibility with UNIX®[*] systems.

protocol
Must be `tcp` for the TCP protocol. It is mandatory to enter this field for compatibility with UNIX systems.

(no)wait
Must be `nowait` for the TCP protocol. It is recommended to enter this field for compatibility with UNIX systems.

user
User ID to be used when starting INETD. This entry is not currently used. It is recommended that you enter `root` here for compatibility with UNIX systems.

[*] UNIX® is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

server

The file name of the server program, suitable for RPing, or the word "internal" for the time service. For example, smtp, smtp.run, or /programs/smtp.run.

-flags

A string of *flag* characters and <*arguments*>, preceded by a dash (-). *Flags* are single characters; <*arguments*> are blank-separated strings consumed in left-to-right order as the *flags* requiring <*arguments*> are encountered. There may be more than one *flag* string given, or none at all. *Flags* are:

c — Enable TCP checksumming on the connection.

l — Log connections on this service to the file /etc/inetd.log.

q <*n*> — Set the maximum connection queue size for this service to <*n*>, an integer from 0-5. The default is 5.

r <*n*> — Set the maximum receive size in bytes to <*n*>, an integer from 1-8000. The default is 100.

s <*n*> — Set the maximum send size in bytes to <*n*>, an integer from 1-8000. The default is 100.

The following three flags strings are equivalent:

```
-lqr 1 200
-l -q 1 -r 200
-rlq 200 1
```

runstring

The runstring to send to the server, starting with the program name. For example, soivr -r will pass the string XQ,SOIVR,-R. The <*arguments*> have the following substitution strings:

%n — Replaced by a unique 16-character name for the VC socket, which will be given away with this name via IPCGive. See the discussion below on the connection passing mechanism supported.

%% — Replaced by a single %, used to specify the percent character and not a substitution string.

Discussion

For the HP supplied services, FTP, TELNET, and SMTP, the only fields that users may modify are *server* and *flags*. The *server* field should be changed if the server program file is not in /PROGRAMS. Users may also add l to the *flags* parameter to enable logging for that service.

Security and Logging Restrictions

Security and logging restrictions are `inetd.conf` lines that do not begin in column 1. These follow a service definition as described in the “Services Definition” subsection. The format is shown below.

Syntax

```
<keyword> <host/net addresses>
```

Parameters

<keyword> One of the following identifiers:

`ALLOW`	To restrict use of the service to only the given hosts/networks
`DENY`	To forbid use of the service by the given hosts/networks
`NOLOG`	To inhibit logging connections by the given hosts/networks.

<host/net addresses> A list of host or network IP addresses to which the *<keyword>* applies, separated by blanks or commas. For example, `15.8.8.3` specifies the host with this IP address.

Each field may also be given as a range, so `15.8.8.1-10` refers to nodes 1 through 10 on the `15.8.8` network. The `*` wildcard can be used to specify entire (sub)networks, so `15.8.8.*` refers to every host on the `15.8.8` network. This is the same as simply entering `15.8.8`.

Example

The `inetd.conf` entry below defines a service at TCP port `500` which is serviced by program `soivr`. Access to this service is restricted to hosts on the `15.8.10` and `15.8.12` networks.

```
500 stream tcp nowait root soivr soivr %n
   allow  15.8.10  15.8.12
```

The next example defines a service named `aloha` in `/etc/services`, with logging enabled and both send and receive sizes of `500` bytes. The runstring used will be `aloha donho` followed by the name for the VC socket descriptor. Access is denied to host `26.0.0.73` and any host on the `192.6.47` network. Connections from the local net `15` will not be logged.

```
aloha stream tcp nowait root aloha -lsr 500 500 aloha donho %n
   deny  26.0.0.73 192.6.47
   nolog 15
```

If changes are made to `inetd.conf`, then INETD must be restarted in order for the changes to take effect.

Services Mapping

The /etc/services file maps service names to the TCP port the services use. In this file, a line cannot start with a space. Items are separated by any number of blanks. A slash (/) is used to separate the port and protocol name (for example, 21/tcp). A pound (#) sign indicates the beginning of a comment. Characters from the pound (#) sign to the end of the line are not interpreted by routines that read the file. For each service, a single line should be present as shown below.

Syntax

```
officialname  portnumber/protocol [aliases]
```

Parameters

officialname The official name of the service, 1-16 characters.

portnumber The protocol port number for the service, 1-32767. Ports 1-255 are reserved by the Internet Network Information Center for “well-known” services (such as SMTP).

protocol Must be tcp for the Transmission Control Protocol.

aliases Alternate names by which the service is known.

Example

```
#service_name   port/protocol
#------------   -------------
ftp              21/tcp               # File Transfer Protocol
telnet           23/tcp               # TELNET virtual terminal
smtp             25/tcp      mail     # Simple Mail Transfer Protocol
printer         515/tcp   spooler     # Remote print spooling
```

Using INETD with User-Written Server Programs

INETD can handle incoming connections for user-written NetIPC server programs as it does for HP supplied servers. INETD can listen for up to 30 TCP protocol addresses (or “ports”), requiring fewer system resources than if separate programs were to listen for these connections.

Note INETD cannot pass connections to servers using BSD IPC.

INETD `RP`s and schedules a copy of the appropriate server program for each incoming connection from a client program. Thus it is ideal for network applications that require a separate copy of the server program for each client connection. It is also a good idea to use INETD for infrequently used servers. This way the networking resources and the ID segment required by the server are only allocated when the service is actually being used.

For user-written servers, the string `%n` must be part of the runstring specified in the INETD configuration file. After receiving the connection, INETD creates a unique 16 character name that is temporarily assigned to the VC socket descriptor by a call to `IPCGive`. This socket name replaces the `%n` in the runstring used to schedule the server program.

For example, runstring `soivr %n` will cause INETD to execute runstring `XQ,SOIVR,INET3AB423433F3` and give away the VC descriptor with the name starting with `INET`.

The server program must call `IPCGet` with the socket name created by INETD to obtain the VC socket descriptor. At this point, the connection is fully established and the server can send and receive data. Using the VC socket descriptor obtained from `IPCGet`, the server program can call `IPCSend` and `IPCRecv`. There is no need to use `IPCCreate`, `IPCConnect`, or `IPCRecvCn`.

See the NetIPC section in the *NS-ARPA/1000 User/Programmer Reference Manual*, part number 91790-90020, for full information about NetIPC calls.

Example

The following example program section shows how the socket name can be obtained from the runstring and used in the `IPCGet` call. Note that the *`nlen`* parameter in the `IPCGet` call must be 16.

```
ftn7x
      program soivr
      implicit none

      integer*4  vcdesc, result
      integer*2  sockname_i(8)
      character  sockname_c*16
      equivalence(sockname_i, sockname_c)

      sockname_c = ' '
      call fparm(sockname_c)

      call ipcget(sockname_i, 16j, 0j, vcdesc, result)


               .
               .
               .
```

11

NRINIT, Nodal Registry Configuration

The Nodal Registry is an area in DSAM that contains information that the transport and services use to establish connections with remote nodes. After initializing NS-ARPA with NSINIT, initialize the Nodal Registry by running the NRINIT program. You need to initialize the Nodal Registry only if one or more of the following is true:

- You need to communicate with Ethernet-only nodes.
- You need to communicate with non-HP nodes.
- You need to communicate with nodes on another LAN. However, if you have a Probe Proxy Server on the same LAN, then initializing the Nodal Registry is not needed.

With an IEEE 802.3 LAN-only network, NS-ARPA will probe for node names that are not in the Nodal Registry. If found, the node name is added to the Nodal Registry.

The Nodal Registry stores its information in *Nodal Path Reports* (NPRs). Nodal Path Reports contain IP addresses, LAN station addresses (optional), and other protocol information. NPRs also contain information on NS Common Services, such as NFT, and on ARPA Services, such as TELNET and FTP. The node name and address information in the NPR of any node must be consistent with the information configured via NSINIT at that node.

The program *NRINIT* initializes the Nodal Registry. NRINIT takes character strings, builds NPRs, and enters the NPRs in the Nodal Registry. The character strings that it uses to build the NPRs are called *raw NPRs*. You can run NRINIT and specify the name of a text file that contains raw NPRs. To create such a file, use a file of raw NPRs with a text editor and follow the syntax for raw NPRs, as described later in this section.

NRINIT also allows you to interactively add NPRs to the Nodal Registry; delete NPRs; verify the syntax of raw NPRs; and compare raw NPRs with the NPRs in the Nodal Registry. In addition, there is an NS-ARPA/1000 utility to print the contents of the Nodal Registry, called NRLIST. You can use the output from NRLIST as input for NRINIT. NRLIST is documented in *NS-ARPA/1000 Maintenance and Principles of Operation*.

In summary, you must perform the following tasks to initialize the Nodal Registry:

- (Optional) Create a text file with raw NPRs, using a text editor. You can also use the Nodal Registry List utility, NRLIST, to prepare the file. NRLIST is documented in *NS-ARPA/1000 Maintenance and Principles of Operation*.
- Run NRINIT, specifying the text file with raw NPRs or entering the raw NPRs interactively.

This section contains the following subsections:

- Nodal Path Reports
- NRINIT

Nodal Path Reports

The local node is automatically put into the Nodal Registry by NSINIT. Refer to the "Network Management" subsection in Section 9, "NSINIT," earlier in this manual. If you try to configure an NPR for the local node, NRINIT issues a warning message and ignores that NPR.

You must configure Nodal Path Reports as follows:

- If the local node does not have a LAN LI, you must configure Nodal Path Reports for all the nodes in the *internetwork*, except any nodes that do not support NS Common Services or ARPA Services.
- If the LAN does not have a Probe Proxy Server or the local node is a Probe Proxy Server, you must configure NPRs for non-HP, non-802.3 nodes, and nodes on another LAN to communicate with these other nodes.
- If the local node is a member of a LAN with a Probe Proxy Server and is not a Probe Proxy Server, configuring NPRs for remote nodes is optional. The local node can use Probe for an IEEE 802.3 LAN or ARP for an Ethernet LAN to query the Probe Proxy Server for NPRs. However, using Probe to get an NPR takes more time than looking up a configured NPR.

  For more information on Probe, refer to the Section 3, "Network Configuration."

Uniqueness

NPRs must be unique according to node names; the Nodal Registry will only accept one NPR per node name. Therefore, a node with one name but several IP addresses requires one NPR that has multiple addresses. A node with several names requires an NPR for each different name.

Raw NPRs

Raw NPRs are character strings that NRINIT uses to build NPRs. A raw NPR contains a node name and an IP address (or addresses). If the node has a LAN link, the raw NPR can also contain a LAN station address (or addresses).

There are two types of raw NPRs: single address and multiple address.

- Single address: for nodes that have only one IP address (i.e., it belongs to only one network).
- Multiple address: for nodes that have more than one IP address (i.e., it belongs to more than one network).

Single Address Raw NPRs

Single-address raw NPRs are used for nodes that have one IP address per node name.

Syntax

```
BEGIN nodeName IPAddress IP [LIType STAddress] END
```

Parameters

BEGIN — Literal keyword indicating the beginning of the entry.

nodeName — The remote node’s name. This can be a fully-qualified node name or the *name* field only. If you specify the *name* field only, the *domain* and *organization* fields default to the local node’s domain and organization fields. For more information on node names, refer to the “Network Configuration” section.

The local node is automatically put into the Nodal Registry by NSINIT. Refer to the “Network Management” subsection in Section 9, “NSINIT,” earlier in this manual. If you try to configure an NPR for the local node, NRINIT issues a warning message and ignores that NPR.

IPAddress — The remote node’s IP address. For more information on IP addresses, refer to “Names and Addresses” in the “Network Configuration” section.

IP — Literal field.

LIType — The link interface (LI) type of the remote node. Possible link types are:

802	IEEE 802.3 link type
ETHERNET	Ethernet link type
LAN	Both IEEE 802.3 and Ethernet link type

STAddress — The node’s LAN station address on the network indicated by IPAddress.

The *LIType* and *STAddress* parameters are optional. If you omit them, the LAN software will broadcast an address request message and dynamically figure out the remote node’s station address.

For more information on LAN station addresses, refer to Section 3, “Network Configuration.”

END — Literal keyword indicating the end of the entry. For a multiple address node, only one END is specified after all the address entries.

Example of Single Address Node Entries

```
    BEGIN CLARK.NS.HP1K 192.006.001.003 IP 802 08-00-09-60-70-AF END
    BEGIN ANNE.NS.HP1K 192.006.001.002 IP 802
    08-00-09-82-43-C4 END
    BEGIN ALEX.NS.HP1K 192.006.001.001 IP END
    BEGIN LISSA.NS.HP1K 192.006.001.009 IP END
```

Multiple Address Raw NPRs

Multiple-address raw NPRs are used for nodes that belong to more than one network (nodes that have more than one IP address per node name). The format is similar to the single address node entries, except you use a single END literal to indicate the end of the node entry. A raw NPR can be longer than one line, but the NPR constructed from it must have no more than 500 characters (bytes). Each entry containing a node name, IP address, and LAN station address adds approximately 20 characters (bytes) to the NPR. You cannot split a field across a line boundary.

Syntax

```
    BEGIN nodeName IPAddress IP [LIType STAddress]
    [BEGIN nodeName IPAddress IP [LIType STAddress]  ]... END
```

The parameter meanings are the same as those for single-address NPRs.

Duplicate Entries

Note that the following two cases are not the same:

```
    Case 1:
    BEGIN chee.ind.hp 192.006.001.001 IP
    BEGIN chee.ind.hp 192.006.250.002 IP END

    Case 2:
    BEGIN chee.ind.hp 192.006.001.001 IP END
    BEGIN chee.ind.hp 192.006.250.002 IP END
```

The first case is an NPR for the node *chee.ind.hp* that has two addresses; the second case is a set of two NPRs for the node *chee.ind.hp*. In the second case, the Nodal Registry will consider the two raw NPRs to be duplicates.

Example of Multiple Address Node Entries

The node SUPER.BIG.NODE belongs to three networks and has three IP addresses:

```
BEGIN SUPER.BIG.NODE 192.006.001.001 IP
BEGIN SUPER.BIG.NODE 192.006.250.002 IP
BEGIN SUPER.BIG.NODE 192.006.251.003 IP END
```

NRINIT

NRINIT takes raw NPRs, builds NPRs and enters them in the Nodal Registry. NRINIT also allows you to

- interactively enter NPRs in the Nodal Registry
- delete NPRs
- verify the syntax of raw NPRs
- compare raw NPRs with the NPRs in the Nodal Registry

The NS-ARPA/1000 subsystem must be initialized (via NSINIT) before you run NRINIT. In addition, you must have RTE superuser capability to run NRINIT.

Duplicate NPRs

If you attempt to add an NPR to the Nodal Registry that is a duplicate of an existing NPR, NRINIT will either replace the existing NPR or ignore the new NPR, depending on its mode of operation. NRINIT considers an NPR to be a duplicate of another NPR if the NPRs have the same fully-qualified node name but different address information or address information in a different sequential order. If an NPR has the same name and address information as another NPR, NRINIT will replace the existing NPR or ignore the second NPR, depending on its mode of operation.

Local Node

The local node is automatically put into the Nodal Registry by NSINIT. Refer to the "Network Management" subsection in Section 9, "NSINIT," earlier in this manual. If you try to configure an NPR for the local node, NRINIT issues a warning message and ignores that NPR. This feature allows you to keep the same NPR text file without penalizing you for having a local node name entry.

NRINIT Runstring

```
[RU,]NRINIT[,inputDevice][,logDevice][,mode]
```

Parameters

inputDevice Device (file or interactive device LU) to provide input for NRINIT.

If *inputDevice* is a file, NRINIT will open that file and read it for input, as described in the following subsection "Input Files." NRINIT treates any line with an asterisk (`*`) in the first column as a comment. You can use the NRLIST program, described later in this section, to create an input file for NRINIT.

If *inputDevice* is an interactive device LU, then NRINIT will prompt the user for input, as described in the subsection “Interactive Operation.”

Default: local LU 1 (scheduling terminal).

logDevice Device (file name or device LU) to which NRINIT will log any messages or errors.

Default: local LU 1 (scheduling terminal).

mode The user may choose one or none of the following.

C Indicates Comparison mode. In this mode, NRINIT compares raw NPRs with the Nodal Registry’s contents. NRINIT prompts you for raw NPRs if *inputDevice* is an interactive LU; if *inputDevice* is a file, NRINIT reads the file for raw NPRs. NRINIT prints a report if it encounters any duplicate entries.

V Indicates Verify mode. In this mode, NRINIT will check the syntax of raw NPRs but will not change the Nodal Registry’s contents. NRINIT prompts you for raw NPRs if *inputDevice* is an interactive LU; if *inputDevice* is a file, NRINIT reads the file for raw NPRs.

D Indicates Delete mode. In this mode, NRINIT deletes NPRs from the Nodal Registry with node names that match those that you specify. NRINIT prompts you for fully-qualified node names if *inputDevice* is an interactive LU; if *inputDevice* is a file, NRINIT reads the file for fully-qualified node names. You cannot delete the NPR for the local node.

P Indicates Permission mode. In this mode, NRINIT will build NPRs from raw NPRs. NRINIT prompts you for raw NPRs if *inputDevice* is an interactive LU; if *inputDevice* is a file, NRINIT reads the file for raw NPRs. If the new NPR is a duplicate of an existing NPR in the Nodal Registry, NRINIT will prompt you at the scheduling terminal, asking if you want to replace the existing NPR with the new one. If the new NPR is not a duplicate, NRINIT will add it to the Nodal Registry.

N Indicates Non-Duplicate mode. In this mode, NRINIT will build NPRs from raw NPRs and will add only the NPRs to the Nodal Registry that are not duplicates of existing NPRs. NRINIT prompts you for raw NPRs if *inputDevice* is an interactive LU; if *inputDevice* is a file, NRINIT reads the file for raw NPRs.

R Indicates Replace mode. In this mode, NRINIT will parse raw NPRs and will add the NPRs to the Nodal Registry. If an NPR is a duplicate of an existing NPR, NRINIT will replace the existing NPR. NRINIT prompts you for raw NPRs if *inputDevice* is an interactive LU; if *inputDevice* is a file, NRINIT reads the file for raw NPRs. You cannot replace the NPR for the local node.

Input Files

If you specify a file for *inputDevice* in the runstring and the NRINIT operation mode is Comparison, Verify, Delete, Permission, Non-Duplicate, or Replace, then the file must be a text file with raw NPRs. If the operation mode is Delete, the file must contain the fully-qualified names of the nodes that NRINIT is to delete.

NRINIT will treat any line with an asterisk (`*`) in the first column as a comment.

A sample NRINIT answer file is in `/NS1000/EXAMPLES/ALL_NODES.NRIN` and is shown in Appendix A.

Error Handling

If you enter an inappropriate response and *inputDevice* is interactive, NRINIT will print an error message and re-prompt you until you enter an acceptable response.

If you enter an inappropriate response and *inputDevice* is not interactive, NRINIT will print an error message to *logDevice* and terminate.

If NRINIT terminates with an error, it will return an error code to its scheduling program in PRTN parameter $RETURN1; otherwise, it will return a zero. For information on PRTN parameters, refer to the *RTE-A User's Manual* and *RTE-A Programmer's Reference Manual*. NRINIT error messages are documented in the *NS-ARPA/1000 Error Messages and Recovery Manual*.

NRINIT Mode Menu

If you did not specify an NRINIT operation mode in the runstring, NRINIT prints the following menu and prompts you for the mode:

```
Mode selection. Please choose one of the following:
C   Comparison mode
V   Verify mode
D   Delete mode
--The following modes add to Nodal Registry--
P   ask for Permission to replace duplicates
N   add only Non-duplicates
R   Replace duplicates without asking for permission

Enter /E to exit
NRINIT>
```

Enter the letter that corresponds to the NRINIT operation mode that you want, or enter `/E` to exit. (Refer to NRINIT's runstring parameter descriptions for more information on the modes.)

NRINIT Interactive Operation

If you specified an interactive device LU for *inputDevice*, NRINIT prints the following banner to your terminal:

```
Nodal Registry Configuration program
Parameters:runString
<revisionDate>
date time
Version 1.0 Mode: mode
At node nodeName
```

Where

runString The NRINIT runstring you entered and any parameters you specified there.

revisionDate Software revision date.

date time Current date and time.

mode The single letter that corresponds to the mode you selected.

nodeName Name of the node at which you are running NRINIT.

Comparison, Non-Duplicate, Permission, Replace, and Verify Modes

If NRINIT's operation mode is Comparison, Non-Duplicate, Permission, Replace, or Verify, then NRINIT prints the following raw NPR description and prompts you for a raw NPR:

```
*++ Nodal Registry ++*
* Enter a nodal path report for each node in the network.
* format: BEGIN <node name> <IP addr> <protocol descr> END
*
* Where:
* node name = name.domain.organization (each field 1..16 char)
* IP addr = IP address n.n.n.n, where n = 0.255
*
* Protocol description format (choose 1):
*
* IP
* IP 802/ETHERNET/LAN <LAN addr>; where lan addr HH-HH-HH-HH-HH-HH
*                           H = hexadecimal digit
*
NRINIT>
```

Enter a raw NPR or `/E` to exit.

If you enter a raw NPR with a node name that is not fully-qualified, NRINIT prints a warning message that indicates the fully-qualified node name (*node.domain.organization*) that it will use:

```
Warning:  defaulted domain and/or organization subfield.
Taking full environment as:  name.domain.organization
```

If you enter an illegal raw NPR, NRINIT will re-prompt you with the following message:

```
You may start over again at BEGIN or terminate with /E.
```

Enter another raw NPR, starting with the `BEGIN` parameter, or enter `/E` to exit.

If you enter a valid raw NPR, NRINIT will process it according to NRINIT's operating mode, as described in the following subsections.

You cannot replace or delete the NPR for the local node. NRINIT issues a warning and ignores the entry.

Comparison (C) Mode

In Comparison mode, NRINIT prompts you for raw NPRs until you enter `/E`. NRINIT then prints the number of raw NPRs that have the same fully-qualified node name but different address information as existing entries in the Nodal Registry. If an NPR has multiple addresses but the order of the address information does not match, NRINIT will consider the NPRs to be different.

```
nn differences were found.
```

If there are no cases where a Nodal Registry entry's name matches a name in a raw NPR, or if in all such cases the address information also matches, then NRINIT prints the following message:

```
No differences were found.
No new nodes were encountered.
```

NRINIT prints the following message before terminating:

```
There are xxxx entries in the Nodal Registry.
```

The value *xxxx* is the total number of NPRs in the Nodal Registry.

Non-Duplicate (N) Mode

In Non-Duplicate mode, NRINIT builds a new NPR from the raw NPR. If the new NPR is not a duplicate of any NPRs in the Nodal Registry, NRINIT enters the new NPR in the Nodal Registry. If the new NPR is a duplicate of an NPR in the Nodal Registry, NRINIT will not add the NPR to the Nodal Registry and will print the following message to the `logDevice`:

```
Warning: Duplicate name: nodeName
```

NRINIT will reprompt you for raw NPRs until you enter `/E` to exit. NRINIT will then print the following message:

```
nn entries were added
```

The value *nn* is the number of NPRs that you added.

NRINIT prints the following message before terminating:

```
There are xxxx entries in the Nodal Registry.
```

The value *nn* is the number of NPRs that you added or replaced, and *xxxx* is the total number of NPRs in the Nodal Registry.

Permission (P) Mode

In Permission mode, NRINIT builds a new NPR from the raw NPR. If the new NPR is not a duplicate of any NPRs in the Nodal Registry, NRINIT enters it in the Nodal Registry. If the new NPR is a duplicate of an NPR in the Nodal Registry, NRINIT will print the following message to the `logDevice`:

```
Warning: Duplicate name: nodeName
```

Then, NRINIT asks if you want to replace the existing NPR:

```
OK to replace [Y/(N)]?
```

Enter `Y` if you want NRINIT to replace the existing NPR with the new NPR; enter `N` if you want NRINIT to ignore the new NPR.

If you enter `Y`, NRINIT replaces the existing entry and writes the following message to your terminal:

```
Overwriting previous entry.
```

If you enter `N`, NRINIT ignores the new NPR and writes the following message to your terminal:

```
New entry being ignored.
```

NRINIT will re-prompt you for raw NPRs until you enter `/E` to exit. If NRINIT replaced any NPRs in the Nodal Registry, it will print the following message:

```
WARNING: nn duplicate names were detected
 and their configuration information replaced.
This information should be verified
```

The value *nn* is the number of duplicate entries that NRINIT replaced.

NRINIT prints the following message before terminating:

```
nn entries were added
There are xxxx entries in the Nodal Registry.
```

The value *nn* is the number of NPRs that you added or replaced, and *xxxx* is the total number of NPRs in the Nodal Registry.

You cannot replace or delete the NPR for the local node. NRINIT issues a warning and ignores the entry.

Replace (R) Mode

In Replace mode, NRINIT builds an NPR from the raw NPR and enters it in the Nodal Registry. If the new NPR is a duplicate of an NPR in the Nodal Registry, NRINIT prints the following message to the *logDevice*:

```
Warning: Duplicate name: nodeName
Overwriting previous entry.
```

NRINIT will re-prompt you for raw NPRs until you enter `/E` to exit. If NRINIT replaced any NPRs in the Nodal Registry, it will print the following message:

```
WARNING: nn duplicate names were detected
 and their configuration information replaced.
This information should be verified
```

The value *nn* is the number of duplicate entries that NRINIT replaced.

NRINIT prints the following message before terminating:

```
nn entries were added
There are xxxx entries in the Nodal Registry.
```

The value *nn* is the number of NPRs that you added or replaced, and *xxxx* is the total number of NPRs in the Nodal Registry.

You cannot replace or delete the NPR for the local node. NRINIT issues a warning and ignores the entry.

Verify (V) Mode

In Verify mode, NRINIT checks the syntax of the raw NPR. If there are any errors, NRINIT prints an error message and re-prompts you with the following message:

```
You may start over again at Begin or terminate with /E
```

NRINIT will re-prompt you for raw NPRs until you enter `/E` to exit.

NRINIT prints the following message before terminating:

```
There are xxxx entries in the Nodal Registry.
```

The value *xxxx* is the total number of NPRs in the Nodal Registry.

Delete (D) Mode

If the operation mode is Delete, then NRINIT will prompt for the name of the node for which you want to delete the NPR:

```
Name to delete [(/E)]:
```

Enter the fully-qualified name of the node for which you want to delete the NPR, or enter RETURN or `/E` to exit NRINIT.

If NRINIT successfully deletes the NPR, it prints the following message:

```
nodeName deleted
```

If NRINIT cannot find any NPRs for the node name that you specified, it prints the following message:

```
nodeName not found
```

NRINIT will keep prompting you for a node name until you enter `/E` in response to its prompt. After you enter `/E`, NRINIT will print the following message if any NPRs were deleted:

```
nn entries were deleted.
```

Otherwise, NRINIT prints the following message:

```
No entries were deleted.
```

NRINIT prints the following message before terminating:

```
There are xxxx entries in the Nodal Registry.
```

The value *xxxx* is the total number of NPRs in the Nodal Registry.

You cannot replace or delete the NPR for the local node. NRINIT issues a warning and ignores the entry.

Example

```
CI> nrinit
 ** Nodal Registry Configuration Program **
Enter ? for help

 Mode selection.  Please choose one of the following:
 C   Comparison mode
 V   Verify mode
 D   Delete mode
 --The following modes add to Nodal Registry--
 P   Ask for permission to replace duplicates
 N   Add only non-duplicates
 R   Replace duplicates without asking for permission

 Enter /E to exit
 NRINIT> p
 Parameters:RU,NRINIT
 <930907.1159>
 Fri Oct 22, 1993   5:29 pm Version 1.0 Mode:P
  At node GROUCHO1K.HP.COM
 *++ Nodal Registry ++*
 * Enter a nodal path report for each node in the network.
 * format: BEGIN  <node name > <IP addr> <protocol descr> END
 *
 * Where:
 * node name = name.domain.organization (each field 1..16 char)
 * IP addr = IP address n.n.n.n, where n = 0..255
 *
 * Protocol description format (choose 1):
 *
 *   IP
 *   IP 802/LAN/ETHERNET <LAN addr>; where LAN addr HH-HH-HH-HH-HH-HH
 *                              H = hexadecimal digit
 *
 NRINIT> begin eric.ind.hp 192.006.001.001 ip end
 NRINIT> begin eric.ind.hp 192.006.250.002 ip end
 Warning: Duplicate name:ERIC.IND.HP
 OK to replace [Y/(N)]? RETURN
 New entry being ignored.
 NRINIT> begin eric.ind.hp 192.006.251.001 ip end
 Warning: Duplicate name:ERIC.IND.HP
 OK to replace [Y/(N)]? y
 Overwriting previous entry.
 NRINIT> /e
 WARNING:     1 duplicate names were detected
 and their configuration information replaced.
 This information should be verified
 1 entry was added
 There are     2 entries in the Nodal Registry

CI> nrinit
 ** Nodal Registry Configuration Program **
 Enter ? for help

 Mode selection.  Please choose one of the following:
 C   Comparison mode
 V   Verify mode
 D   Delete mode
 --The following modes add to Nodal Registry--
 P   Ask for permission to replace duplicates
 N   Add only non-duplicates
 R   Replace duplicates without asking for permission

 Enter /E to exit
 NRINIT> d
 Parameters:RU,NRINIT
 <930907.1159>
 Fri Oct 22, 1993   5:37 pm Version 1.0 Mode:D
  At node GROUCHO1K.HP.COM
 Name to delete [(/E)]? eric.ind.hp
 eric.ind.hp deleted
 Name to delete[(/E)]? /e

 1 entry was deleted
 There is one entry in the Nodal Registry
CI>
```

12

Additional Initialization Tasks

This section describes additional initialization tasks that you must perform after initializing NS-ARPA with NSINIT. These tasks are summarized below and explained in more detail in the subsequent subsections.

- You must initialize TELNET pseudo terminal LUs every time you reboot. If you have TELNET pseudo terminal LUs generated in your system, you must put these LUs in the `HPMDM_TABLE` table before they are used by TELNET. `HPMDM_TABLE` is a table containing a list of modem LUs. The TELNET pseudo terminal LUs are linked to these modem LUs so that they can be identified by HPMDM for clean up purposes.

  Up to eight such LUs can be specified. If you need to increase this amount, refer to "Changing HPMDM_TABLE" later in this section.

- If you have Remote I/O Mapping, you must initialize it before using it.

- If you have a Bisync link and you want to initiate traffic to the HP 3000, you must initialize the link by running DSLIN. If your node only needs to receive data from the HP 3000 (i.e., it will never need to initiate traffic), you do not need to run DSLIN; the `DSLINE` command should be entered at the HP 3000 instead.

- If you want to use the RMOTE `MO` (MOVE) command to transfer files to an HP 3000 that supports DS/1000-IV Compatible Services, you must install the slave program, `COPY3K.PUB.SYS` on the HP 3000. You should re-install `COPY3K.PUB.SYS` each time you re-install NS-ARPA/1000 software.

- If you want to use the HP 1000 file server program that services file access requests from remote NFS* clients, run FSRV. For details, refer to the *File Server Reference Guide for NS-ARPA/1000 and ARPA/1000*, part number 91790-90054.

* NFS is a trademark of Sun Microsystems, Inc.

TELNET Pseudo Terminal LUs

If you have TELNET pseudo terminal LUs generated in your system, you must put these LUs in the `HPMDM_TABLE` table before they are used by TELNET. `HPMDM_TABLE` is used by RTE to identify the modem LUs and TELNET pseudo terminal LUs. Up to eight LUs can be initialized using the default `HPMDM_TABLE`. This amount can be changed as explained later in this section in “Changing HPMDM_TABLE.”

`HPMDM_TABLE.REL` is generated into Labeled System Common first and then initialized by running the program `HPMDM`. It is recommended to run `HPMDM` in the system bootup welcome file for convenience. `HPMDM` is also documented in the RTE-A manuals. Use the following syntax to initialize the TELNET pseudo terminal LUs.

Syntax

```
RU,HPMDM[,lu,AD[,ok]]
```

Parameters

lu The TELNET pseudo terminal LU specified in the system generation.

`AD` Add the TELNET pseudo terminal LU to the `HPMDM_TABLE` table.

`ok` Initialize the LU even if it is already in the table.

Discussion

If no parameters are specified for `HPMDM`, it will display a help screen to provide you with more information.

The following example shows you how to put pseudo terminal LUs 30, 31, 95, and 96 in `HPMDM_TABLE`:

```
RU,HPMDM,30,AD,ok
RU,HPMDM,31,AD,ok
RU,HPMDM,95,AD,ok
RU,HPMDM,96,AD,ok
```

TELNET can then use the above pseudo terminal LUs to establish a virtual terminal connection.

The maximum number of pseudo terminal LUs that can be used by TELNET is 64. If more are defined in the system generation and in `HPMDM_TABLE`, only up to 64 LUs can be used.

Changing HPMDM_TABLE

Up to eight pseudo terminal LUs and modem LUs can be specified in the system generation and initialized by HPMDM. This amount can be changed by changing the source file for HPMDM_TABLE. The source file is /RTE_A/HPMDM_TABLE.MAC. A partial listing of this source file is listed below.

```
Macro,q,c
  list medium

* Change this value to the sum of the number of modems and the number of
* TELNET LUs in the system.

&NumberOfPorts   iglobal 8            ; number of LUs in the table

*     NAME   : HpMdm_Table
*     SOURCE : 92077-18940  System common table for HpMdm
*     RELOC  : 92077-16940
      .
      .
      .
```

The sequence of steps to change the number of pseudo terminal LUs is as follows:

1. Change &NumberOfPorts from eight to the number of LUs that you want.
2. Assemble the source file with MACRO/1000, which generates a revised HPMDM_TABLE.REL file.
3. Generate the revised HPMDM_TABLE.REL file into Labeled System Common.
4. Run HPMDM (after initializing NS-ARPA and before using TELNET) to initialize the pseudo terminal LUs.

Remote I/O Mapping

To initialize Remote I/O Mapping, enter the following command in response to a CI prompt or in the system WELCOME file:

```
[RU,]IOMAP,mappableLU,-1
```

The *mappableLU* is any mappable LU.

For more information, refer to the section “Remote I/O Mapping” in this manual.

TELNET is not supported on a terminal that has been mapped to a remote HP 1000 node with remote I/O mapping.

Initializing Bisync Connections

Before you can begin sending NS-ARPA messages over a Bisync link, you must initialize the link by running the NS-ARPA/1000 program DSLIN at the HP 1000 or by issuing the `DSLINE` command at the HP 3000.

The following paragraphs describe how to use the NS-ARPA/1000 program DSLIN to initialize a Bisync link. If your node only needs to receive data from the HP 3000 (i.e., it will never need to initiate traffic), you do not need to run DSLIN; the `DSLINE` command should be issued at the HP 3000 instead. For more information on the `DSLINE` command, refer to the *NS/3000 User/Programmer Reference Manual*, part number 32344−90001.

Note Both the HP 1000 and the HP 3000 Bisync cards must be enabled before initialization messages can be exchanged. On the HP 1000, the card is enabled when NSINIT initializes NS-ARPA/1000. On the HP 3000, the `DSCONTROL` command must be issued.

Using DSLIN to Initialize a Link

When DSLIN is issued to initialize a Bisync link, it determines

- the HP 3000 LU number to initialize
- the communications block size to use
- the maximum retry count
- the connect timeout
- the local and remote ID sequences
- the connect mode (primary or secondary) of the link.

These characteristics may be provided as parameters in the DSLIN runstring, as answers in an answer file, or as responses to interactive DSLIN prompts.

Note DSLIN is scheduled automatically (if it has an ID segment) when the `HELLO` command or call is used at the HP 1000 and the Bisync link is not initialized. When DSLIN is scheduled automatically, the link is initialized in primary mode with the default DSLIN line characteristics. (These line characteristics are described later in this section.)

DSLIN

Use the following syntax to initialize a Bisync link with DSLIN:

Syntax

```
                 ┌   ┌,S┐             ┐
RU,DSLIN[,lu]    │,- └,P┘             │
                 └,answerFile         ┘
```

Parameters

lu This is the transmit LU of the transmit/receive pair of the Bisync link.

Default: If this parameter is omitted, DSLIN will prompt you for the logical unit number.

answerFile An answer file that contains responses to the DSLIN prompts. (The DSLIN prompts are listed below.)

Default: If this parameter is omitted and "`-`" is also not specified, DSLIN will prompt you for line characteristics.

`-` Indicates that the line will be opened in primary mode with the following DSLIN default line characteristics:

- Communications Block Size = 1024 words
- Maximum Retry Count = 8
- Connect Timeout = 255 seconds
- Local and Remote ID Sequences = as specified at NS-ARPA initialization

Default: If this parameter is omitted and *answerFile* is also not specified, DSLIN will prompt you for the line characteristics.

`S` Indicates that the link should be initialized in secondary mode. If the link is placed in secondary mode, the HP 1000 will wait for the HP 3000 to initiate communication. This mode is useful if you have a modem connection to the HP 3000 and connection cost is a consideration. (Bisync links are automatically initialized in secondary mode by NSINIT.)

Default: If neither `S` nor `P` is specified, the link will be initialized in primary mode.

P

Indicates that the link should be initialized in primary mode. If the link is placed in primary mode, the HP 1000 will attempt to initiate communication with the HP 3000.

Default: If neither `S` nor `P` is specified, the link will be initialized in primary mode.

DSLIN Dialogue

The following is an example of the DSLIN dialogue. When a DSLIN parameter has a default value, it is displayed in brackets (`[ ]`) as part of the prompt. Type `/D` or press return to indicate that you want to use the default value. Entering `/A` at any time after the LU has been entered will cause DSLIN to terminate.

To use DSLIN interactively, type the following runstring:

```
:RU,DSLIN
```

DSLIN begins by prompting you for the positive HP 3000 LU number:

```
LU OF HP 3000:
```

Note DSLIN will also prompt you for an LU number when an answer file is used. Do *not* include an LU number in the answer file.

If an invalid HP 3000 LU is entered, DSLIN displays the following message and terminates:

```
LU nn IS NOT IN THE 3000 LU TABLE
END OF DSLIN
```

If you do not enter an LU, DSLIN repeats the prompt with the following message:

```
PLEASE ENTER THE LU.
```

Once a valid HP 3000 LU has been entered, DSLIN asks you if you want to open or close the Bisync link:

```
OPEN (DEFAULT) OR CLOSE THIS LU?
```

If you reply `CLOSE`, DSLIN attempts to close the line. Once the line is closed, DSLIN displays the following message and then terminates:

```
LU nn WAS CLOSED
END OF DSLIN
```

The line is not closed until a termination reply is received from the HP 3000. If the HP 3000 does not respond, the line may not be closed.

Note DSLIN offers an alternate way to close a Bisync link. This method is described later in this section in “Using DSLIN to Alter Line Characteristics.”

If you reply `OPEN` (the default response), DSLIN proceeds by prompting you for the communications block size:

```
COMMUNICATIONS BLOCK SIZE [1024] :
```

The communications block size is the maximum size message, including the header and appendage, that NS-ARPA/1000 will attempt to send across the link. You may select a communications block size in any multiple of 16 between 304 and 4096.

Note Although you can specify a communications block size of 4096 words, the actual block size used is limited to the buffer size of the card installed at the HP 3000. HP 3000 Bisync cards may have buffer sizes of either 1024 or 1072 words.

If you respond to this prompt with an out-of-range value, DSLIN displays the following message:

```
INPUT VALUE [nnnn] OUT OF RANGE; MUST BE BETWEEN 304 AND 4096
  DEFAULT VALUE OF 1024 USED
```

If a value which is not a multiple of 16 is specified, DSLIN uses the next highest multiple and prints the following message:

```
BLOCK SIZE CHANGED TO nnnn
```

DSLIN then prompts for the maximum retry count:

```
MAXIMUM RETRY COUNT [16] :
```

Indicate the number of times a message will retry if an acknowledgement fails to arrive. The value must be between 0 and 255.

Next, DSLIN prompts for the response timeout:

```
CONNECT TIMER (SECONDS) [255] :
```

Indicate the time allowed for response before the line is placed in secondary mode. The value must be between 1 and 255. If you respond to this prompt with an out-of-range value, DSLIN displays the following message:

```
INPUT VALUE [nnn] OUT OF RANGE; MUST BE BETWEEN 1 AND 255
  DEFAULT OF 255 USED
```

DSLIN then asks for the local ID sequence:

```
LOCAL ID SEQUENCE :
```

The local ID sequence may consist of up to 15 ASCII characters. If you want a null ID sequence, respond with `/D`. If you do not want a local ID sequence or any remote ID sequences, respond with `/E`.

If you respond to the local ID sequence prompt with a `/D` or an ASCII string, DSLIN prompts you for up to 15 remote ID sequences:

```
REMOTE ID SEQUENCE 1:

REMOTE ID SEQUENCE 2:

      .
      .
      .

REMOTE ID SEQUENCE 15 :
```

Up to 15 ASCII characters are allowed per remote ID sequence. You may also respond with `/D` to indicate a null sequence. Type `/E` to indicate that no more ID sequences are to be used. For multiple HP 3000 connections, each with a unique ID sequence, the values specified here are searched for any match.

At this point in the dialogue, DSLIN sends the configuration information to the board and reads back the board's parameters. If the board type is not 0, DSLIN prints the following message and then terminates:

```
THE BOARD ON LU lu DOES NOT CONTAIN BISYNC FIRMARE!
```

If the board is type zero, DSLIN prompts for the connection mode:

```
INITIALIZE STATION AS PRIMARY OR SECONDARY?
```

You may respond with either `P` or `S` for primary or secondary mode, respectively. (You may also type `/D` to indicate primary mode.)

If you respond with `S`, DSLIN prints the following message and then terminates:

```
CONNECTING AS SECONDARY STATION ON LU lu
BISYNC BOARD ON LU lu REPORTS BUFFER SIZE OF nnnn WORDS
AWAITING CALL ON LU lu
END OF DSLIN
```

If `P` is the response, the link is initialized in primary mode and DSLIN displays the following message:

```
HP 3000 BISYNC LINK ON LU lu READY FOR CONNECTION
BISYNC BOARD ON LU lu REPORTS BUFFER SIZE OF nnnn WORDS
```

DSLIN now goes into a loop where it waits one second, writes a DS/3000 initialization request, and checks the returned I/O status until the message is sent or a timeout occurs. You can prematurely terminate this loop and force a secondary connect by setting DSLIN's break flag with the RTE `BR` command. When DSLIN detects the flag, it prints the following message:

```
BREAK FLAG SET.
```

DSLIN then connects as a secondary station.

If the primary connect request times out (this is usually because the modem connection was not made or, on a direct connection, because the HP 3000 `DSLINE` command was not issued at the HP 3000), DSLIN connects as a secondary station and prints the following message:

```
PRIMARY CONNECT TIMED OUT
```

If the HP 3000 Bisync board is enabled and the HP 1000's primary connect request does not time out, the HP 3000 responds and the link is initialized. DSLIN also displays the following message:

```
    LINE IS UP WITH BUFFER SIZE nnn.
```

If the initialization message is sent but the HP 3000 does not respond within 50 seconds, DSLIN disconnects the link and reconnects as a secondary station. DSLIN also displays the following message:

```
    LINE IS UP BUT 3000 IS NOT REPLYING
```

Using DSLIN to Alter Line Characteristics

After a Bisync link has been initialized with DSLIN, you can rerun DSLIN to alter line characteristics. Primary connections must be placed in secondary mode before their line characteristics can be altered. To place a primary connection in secondary mode, you must close the line by using DSLIN with the following runstring:

```
    RU,DSLIN[,lu][,CLOSE]
```

Parameters

lu The positive HP 3000 logical unit number.

CLOSE Indicates that the line should be closed.

DSLIN will close a Bisync connection only if there are no other users on line. If there *are* other users on the line, DSLIN prints the following message:

```
    SESSIONS STILL OPEN ON LU lu
    THE LU WAS NOT CLOSED.
```

If there are no other users on the line, DSLIN sends a close request to the HP 3000, prints the following message and then terminates:

```
    LU lu WAS CLOSED
```

The line is not closed until a termination reply is received from the HP 3000. If the HP 3000 does not respond, the line may not be closed. If the reply is received, the line closes. The physical link (for example, a modem line) also closes. You can use the NSINF V or L command to verify that the line was closed. (NSINF is documented in the *NS-ARPA/1000 Maintenance and Principles of Operation Manual*.)

Once the line is closed, you can rerun DSLIN with the runstring described in "Using DSLIN to Initialize a Link."

Installing the RMOTE MOve Command Slave Program

The RMOTE MO (MOVE) command is implemented as a Program-to-Program pair; RMOTE is the master program on the HP 1000 and COPY3K.PUB.SYS is the slave program on the HP 3000. To use the MO command, the program COPY3K *must* be installed in the PUB.SYS account. NS-ARPA/1000 supplies the executable file !COPY3, which contains COPY3K, and the program MVCP3, which copies !COPY3 to COPY3K.PUB.SYS on the HP 3000. To install COPY3K, enter the following commands (user input is underlined):

```
CI>RP,MVCP3
CI>RU,RMOTE
$SW
#HELLO MANAGER[/password].SYS[/password]
#SW
$RU,MVCP3[,loggingLU][,filedesc]
```

Parameters

password — The appropriate account or user password for MANAGER.SYS.

loggingLU — The LU to which MVCP3 will report errors.
Default: The local LU 1 (the scheduling terminal).

filedesc — File descriptor of the !COPY3 file. Must be specified if !COPY3 is not in the working directory. !COPY3 is in the directory /NS1000/DSREL. To install from there, use /NS1000/DSREL/!COPY3.

Default: !COPY3.

The following conditions must be met for MVCP3 to be successful:

- You must schedule MVCP3 from RMOTE.
- You must logon to the HP 3000 as MANAGER.SYS, specifying any user or account passwords.
- There must not be an existing copy of COPY3K.PUB.SYS on the HP 3000.

If the above conditions are met, MVCP3 opens !COPY3 and COPY3K.PUB.SYS, and prints the following message:

```
BEGINNING TRANSFER OF !COPY3 TO COPY3K.PUB.SYS
```

MVCP3 then transfers !COPY3 to the HP 3000.

13

Remote VCP (DSVCP) and Forced Cold Loads

Overview

Remote VCP and forced cold loads are services in DS/1000-IV and are part of the DS/1000-IV Compatible Services of the NS-ARPA/1000 product. Therefore, you can use these services between DS/1000-IV and NS-ARPA/1000-IV nodes. References in this section pertain to both networking products with differences noted. NS-ARPA/1000 runs on RTE-A systems only; DS/1000-IV runs on RTE-A and RTE-6/VM systems.

Remote VCP (Virtual Control Panel) allows you to invoke the A-Series VCP of a neighbor node connected by an HDLC link from your node. Remote VCP operates only between adjacent nodes connected by an HDLC link. You can obtain the VCP prompt at your terminal as if you were using a terminal connected to the remote system where VCP is executing. From your node, you can perform the following functions at the remote node:

- boot
- execute the Diagnostic Design Language (DDL)
- modify or examine machine registers
- download programs or an operating system to a memory-based node
- obtain a memory dump

When describing Remote VCP, *master* refers to the computer in control, or the system from which you run DSVCP. *Slave* refers to the remote computer being controlled.

Remote VCP also allows you to set automatic boot from the master node; at power-up, the slave will boot itself up with a system file that resides at a remote node (usually the master). The slave node must be an F- or A-Series node. There are restrictions when using DSVCP to an E- or F-Series. To use DSVCP to an E- or F-Series, you must install the PSI card with HDLC firmware in a select code less than 37B. The Forced Cold Load switch must be enabled on the PSI card and RPL configured/enabled on the CPU board.

You can use DSVCP to control a DS/1000-IV slave. The slave can be an E-, F-, or A-Series CPU, but the only command you can issue to an E- or F-Series is `\BREAK`. When it is sent, the slave halts. If the CPU switch settings are set up for Remote Program Load, the F-Series will automatically reboot (forced cold load).

Remote VCP software communicates with the A-Series VCP using NS-ARPA/1000 or DS/1000-IV link level software and drivers. Remote VCP operates only between adjacent nodes connected by an HDLC link.

The Remote Virtual Control Panel feature is implemented using two programs. The first is the VCP monitor, VCPMN. This program intercepts and displays VCP messages from the slave system and displays them on the system console of the master, or controlling system. VCPMN resides at the master node.

The second program is the VCP master program, DSVCP, which allows access to the control panel of the slave computer, and consequently access and alter its memory, and CPU and I/O registers. This program can access the boot loader programs in the VCP and cause various programs to be downloaded, via the NS-ARPA or DS/1000-IV link, or another loading device. The master program allows the master computer to do a memory dump from the slave to the master computer, and write the dump into a new file.

Both the master program, DSVCP, and the monitor program, VCPMN, reside at the master node.

Note In NS-ARPA/1000, DSVCP works only over HDLC links configured with Router/1000 Link Interfaces. For information on remote VCP and downloading memory-based systems over a LAN link, refer to the *RTE-A System Generation and Installation Manual*.

Required Hardware and Software

The following hardware and software are required to utilize the Remote VCP feature:

- HP 1000 A-Series Computer (slave) installed with a PSI card with HDLC firmware. The PSI card should be configured with the BREAK feature enabled (U1S1 [switch 1] in the closed position). Remember only one card per system can have the Break flag enabled. Refer to the computer reference manual for your machine for CPU switch information.

  The slave computer may also be an F-Series computer with the HDLC card installed in any select code less than 37B. The Forced Cold Load switch should be enabled on the HDLC card and RPL enabled on the CPU board.

- NS-ARPA/1000 or DS/1000-IV Master Node with PSI card with HDLC firmware installed, linked to the slave computer.

- For an autoboot, the select code of the PSI card should be 24. The file downloaded will be the first FMGR file encountered on a system disk with the name `P00000`.

- The required software is as follows:

  DSVCP—VCP master program
  VCPMN—VCP slave monitor
  PROGL—boot file downloader

  VCPMN monitors the link to the A-Series waiting for front panel messages to arrive. When a message arrives, VCPMN displays it on the system console. The operator then runs DSVCP to respond to the A-Series front panel.

- An initialized NS-ARPA/1000 or DS/1000-IV node.

DSVCP Operation

Syntax

```
             ⎧ node ⎫
RU,DSVCP,⎨      ⎬[,DU]
             ⎩ -LU  ⎭
```

Parameters

node Positive node number of link to slave. The neighbor bit in the NR. V must be set. If the neighbor bit is not set, you can use the DSMOD CN command to change this.

-LU Negative LU number of transmit LU to slave.

`DU` Causes an immediate memory dump of the slave. You should only use this option when the VCPMN message `NODE` *`xxx`* `REQUESTING DUMP` is printed at the master's system console, which indicates that the slave node *`xxx`* is soliciting a dump request. If you specify this option when the slave is not soliciting a dump request, VCPMN will return invalid data.

This command schedules the master program, DSVCP. DSVCP then examines the Nodal Routing Vector to verify that the node is a neighbor to the master computer and that it is a legal node. If not, the master program prints:

```
/DSVCP:  ILLEGAL NODE

INPUT DS NODE (CR TERMINATES DSVCP)
```

You can then re-enter the node number or terminate DSVCP by pressing RETURN. If the node is a rerouting node and has not been enabled, the NRV LU entry will be 0. Enter the negative LU number instead of the node number in this case. If you enter the negative LU number, DSVCP tries to get the node number from the NRV. If the node number is not in the NRV, DSVCP will still communicate with the LU; DSVCP uses the node number only as part of message text.

When DSVCP is successfully initiated, it prints a message indicating the node number and LU being used:

```
/DSVCP: NODE xxxxx LU yyy
```

DSVCP then prompts for a command:

```
/DSVCP:
```

During the time that DSVCP is controlling the line between the master computer and the slave computer, the link is taken out of NS-ARPA or DS/1000-IV mode and not available for normal networking traffic.

Normally, after issuing a command to the slave, DSVCP will enter a read mode waiting for some data to be returned. If the command specified a system reboot from the A-Series local disk

(%BDC), no data will be returned. To return DSVCP to normal operation (/DSVCP:), you must set DSVCP's break bit by entering the RTE command BR,DSVCP. In five seconds or fewer, DSVCP will return with a prompt.

DSVCP Commands

If you enter a command preceded by a backslash (\), DSVCP assumes that it is a DSVCP command, and attempts to interpret the command itself. Otherwise, DSVCP sends the command to the slave.

There are five DSVCP commands:

- \BREAK is equivalent to pressing BREAK at the slave's VCP terminal. It halts the slave's CPU.
- \EXIT returns the link to NS-ARPA or DS/1000-IV mode and terminates DSVCP.
- \OPTION turns the VCP prompt (/DSVCP:) off or on.
- \READ sends a read request to the PSI card with HDLC firmware.
- \WAIT causes DSVCP to wait for input from the slave node.

These commands are described below. All the commands can be abbreviated to a backslash followed by the initial letter of the command (\B, \E, \R, \W). If you enter an unrecognizable command, DSVCP prints the following message:

```
/DSVCP: ILLEGAL COMMAND
```

\BREAK (\B)

The \BREAK command sends a BREAK frame to the slave computer via the NS-ARPA or DS/1000-IV driver. This is equivalent to pressing BREAK at the slave's VCP terminal. This halts execution of normal programs at the slave, and transfers control to the Virtual Control Panel at the slave. DSVCP then prints the contents of the slave's P, A, B, M, and T registers at the time of the BREAK. If the slave is an RTE-A, DSVCP also prints the contents of the working map set register.

On the first \BREAK command DSVCP will prompt for the Network Manager's security code:

```
PASSWORD?:
```

DSVCP does not send the BREAK frame until you enter the correct Network Manager's security code (password).

Note This command halts the slave's CPU, and all real-time activity stops. This is why the security code is required at the `PASSWORD?:` prompt.

\EXIT (\E)

The `\EXIT` command returns the link to NS-ARPA or DS/1000-IV mode and terminates DSVCP. Do *not* use the RTE `OF` command to terminate DSVCP. Doing so leaves the link in VCP mode and unusable by NS-ARPA or DS/1000-IV. If there is no DSVCP prompt, use the RTE command `BR,DSVCP`. When DSVCP returns with a prompt, use the `\EXIT` command to exit.

\OPTION (\O)

The `\OPTION` command is used to turn on or off the `/DSVCP:` prompt and the carriage return-linefeed that DSVCP adds to each message read from the slave. In order for multi-line output from the slave VCP to be displayed properly, the carriage return-linefeed must be turned off. The VCP help screen, displayed in response to the `?` command is one example.

The format of the `\OPTION` command is as follows:

```
    \O,prompt,crlf
```

prompt	Zero is off. Non-zero is on. Default is on.
crlf	Zero is off. Non-zero is on. Default is on.

\READ (\R)

The `\READ` command is used to return DSVCP to the NS-ARPA or DS/1000-IV line read mode without sending anything to the slave. This is useful if you have set DSVCP's break bit, but later decide that no data should be sent or if DSVCP was aborted after someone sent a `\BREAK` command. This command causes one read to take place on the receive LU. If the read times out (no message pending) the program requests input from the input console device; otherwise, it prints the incoming message.

You can also use the `\READ` command when messages stack up on the PSI card. One read request retrieves one message. Use the `\READ` command to retrieve any additional messages and to resynchronize DSVCP with the slave computer.

\WAIT (\W)

The `\WAIT` command causes the program to wait for input from the slave node. This command waits for either input from the PSI card or a programmatic BREAK from the operator (`BR,DSVCP`). This command is useful when running the Diagnostic Design Language Interpreter (DDL) remotely via the NS-ARPA or DS/1000-IV link. When one diagnostic completes, you can start another and issue the `\WAIT` command to wait for any completion data.

Downloads

You can use DSVCP to download programs and operating systems to remote memory-based nodes. To do so, you must perform the following tasks:

- Run DSVCP.
- Break into VCP front panel mode at the slave system. To do this, issue the DSVCP `\BREAK` command and then the Network Manager's security code.
- Enter the appropriate VCP command to boot (`%BDS`) or load (`%LDS`) over an NS-ARPA/1000 or DS/1000-IV link.

If you successfully download an operating system, the slave node initializes its DS link (if DINIT was included in the download file); the slave's link stops communication to DSVCP at the master node. DSVCP will be waiting for a reply from the slave's link, which it will never receive. Therefore, you must enter the RTE `BR` command, and set DSVCP's break bit. In five seconds or fewer, DSVCP will issue a prompt, and you can issue the DSVCP `\EXIT` command.

Download Files

Download files must exist on FMGR system cartridges. They must be named `P`*`nnnnn`*, where *`nnnnn`* is an octal number from 00000 to 77777. When you specify a file number in a VCP `%BDS` or `%LDS` command, you enter it in octal (0 to 77777). PROGL converts the octal number to ASCII and uses it to form the `P`*`nnnnn`* file name. For example, if you enter `%LDS15`, PROGL would convert the file number (15) to the ASCII characters 00015 and form the file name, `P00015`. DSVCP would then download the first file named `P00015` that it encounters on a FMGR system cartridge.

Store-and-Forward PROGL

Under normal operation, PROGL would search for the download file on the master node. You can use a store-and-forward version of PROGL, loaded from `%PROGZ`, to download files from other nodes in the network. Store-and-forward PROGL will make remote file access calls to the remote system and ship the file to the node being downloaded.

To use store-and-forward PROGL, you must append a subroutine to PROGL's code that contains the entry points #RMT1 and #RMT2. Store-and-forward PROGL uses these entry points as node numbers to find the download file. PROGL will try #RMT1 and then #RMT2 to find the file.

Example

```
:RU,DSVCP,50
```

Run DSVCP and specify node 50 as the slave.

```
/DSVCP: NODE 50 and LU 88
```

DSVCP confirms communication to node 50 via LU 88.

```
/DSVCP: \BREAK
PASSWORD?: NS
```

Send a BREAK frame to the slave.
NS is the Network Manager's security code.

```
P 003046 A 000000 B 000000 RW 000123 M 003045 T 100020
VCP>
/DSVCP: %BDS110024
```

VCPMN displays the contents of the slave's memory registers, and DSVCP prompts for a command. In this example, the user enters %BDS110024, where:

%BDS tells VCP to boot the system (%B) via a datacomm link (DS)

11 the system file name is P00011

00 placeholder value

24 the PSI card with HDLC firmware has a select code 24B.

```
CM> BR
```

During boot, the slave's datacomm link is initialized; the slave node stops communication with DSVCP at the master node. DSVCP is waiting for a reply from the slave that it will never receive. You must set DSVCP's break bit so that the DSVCP prompt returns.

```
/DSVCP: \EXIT
/DSVCP: END
```

Terminate DSVCP.

Dump Routine

The DSVCP dump routine is used when VCPMN, the monitor program, has indicated that a dump is being requested by the slave A-Series. This might be caused when a user at the A-Series computer wishes to use the NS-ARPA or DS/1000-IV link to capture a memory image for later analysis (also called a crash dump). The user would break into VCP mode at the A-Series locally, or via DSVCP remotely, and enter the command:

```
%WDS
```

DSVCP then prints the following message at the neighbor node:

```
DUMP REQUEST, ENTER NEW FILENAME:
```

Enter the file name at the master node. The file can be a hierarchical file.

When the DUMP routine is initiated, the program requests a file name. The file size specified in the dump file descriptor determines how much memory is dumped. Each block in the file stores 128 words. The default is 256 blocks, which is enough for 32K words of memory. One block is always added to the file size for storing the time of the dump and any comments. The table below can be used as a handy guide for determining file sizes.

Blocks	Pages in K Words	Bytes
256	32	64K
1024	128	256K
4096	512	1M

The dump proceeds with the master sending a request for each block of 128 words. The slave's VCP replies with the memory data. The data is then written to the file, and the memory dump process continues until the file is full.

On the last record, DSVCP prints the following message:

```
TYPE COMMENTS BELOW
```

This allows you to label the file. These comments are contained in the 257th block of the file, with the time and date. This comment record is useful for documenting the conditions of the dump.

If DSVCP encounters an FMP error, it prints the following message:

```
FMP ERROR -xxxxx ON DUMP FILE
```

DSVCP then prompts for another file. If you do not specify a file and only press RETURN, the dump is aborted and DSVCP prints:

```
!?
```

If a line timeout occurs, DSVCP prints the message:

```
LINE TIMEOUT - DUMP ABORTED
```

You could then enter the `\BREAK` command and try again.

A-Series Register Access

To access the A-Series registers, you must first issue the \BREAK command. Then, enter the desired register name into the terminal and press RETURN. DSVCP will print the contents of the specified register as they would appear with local VCP processing. All displays are in octal. To modify the value of the register, enter the new value (in octal) and press RETURN. DSVCP will respond with the register name and the new contents of that register.

Note that the DSVCP will act exactly as if you are communicating directly with the A-Series computer via a local terminal. However, you must terminate all commands by pressing RETURN.

To access a loader, enter the VCP loader command as described in the VCP documentation when communicating with a local terminal. This information is in the computer reference manual for your machine.

Example

```
:RU,DSVCP,8                Run DSVCP and specify node 8 as the slave.

/DSVCP: NODE  8 and LU 84  DSVCP confirms communication to node 8 via LU 84.

/DSVCP: \BREAK             Send a BREAK frame to the slave.
PASSWORD?: NS              NS is the Network Manager's security code.

P 0123247 A 000000 B 000000 RW 000347 M 007745 T 100020
VCP>
/DSVCP: A                  VCPMN displays the contents of the slave's memory
                           registers, and DSVCP prompts for a command. Here, the
                           user enters A, indicating that he wants to modify the
                           contents of the A-Register.

00000
/DSVCP:141652 RETURN       DSVCP displays the contents of the A-Register (00000)
                           and prompts for a command. The user enters the new
                           contents (in octal) and presses RETURN.

A 141652                   DSVCP prints the new contents of the A-Register.
VCP>
/DSVCP:
```

Execution Commands

DSVCP passes the VCP execution commands to the slave's VCP. After an execution command is given, the program issues a read to the NS-ARPA or DS/1000-IV driver. If the read times out, DSVCP checks the break bit in its ID segment (by calling IFBRK). If its break bit is set, DSVCP issues its prompt to the terminal for a command.

If the break bit is not set, DSVCP may wait indefinitely for a VCP message to arrive. This will cause networking traffic between the master and the slave to also be blocked indefinitely. As a result, it is important that you break DSVCP and issue an `\E` command to exit. Do not abort DSVCP, because it is the `\E` command that allows normal networking traffic to commence.

DDL Diagnostic Operation

For operating the DDL (Diagnostic Design Language) via NS-ARPA or DS/1000-IV, the DDL program at the slave uses the VCP protocol when communicating over the NS-ARPA or DS/1000-IV link. DDL operates from the master terminal in the same manner as if the user were talking to the slave A-Series from a local terminal. Because the DDL program uses the VCP protocol for remote operation, you must send a BREAK frame to the slave (to break to front panel mode) before executing DDL.

Example

`CI>RU,DSVCP,6`	Run DSVCP and specify node 6 as the slave.
`/DSVCP: NODE  6 and LU 88`	DSVCP confirms communication to node 6 via LU 88.

`/DSVCP: \BREAK`
`PASSWORD?: NS`
	Send a BREAK frame to the slave.
	`NS` is the Network Manager's security code.

When the break message is received from the A-Series, load DDL which, for this example, is assumed to be in file `P00003` on the master node. In order to download a file, the file must have a file name of `P`*`nnnnn`* where *`nnnnn`* is the file number in the range of 0 to 77777 octal. To initiate a download, only the file number can be passed to the DS Monitor, PROGL, which actually processes the download requests. PROGL must be a scheduled monitor.

The response should be `LC` which means Load Complete. If you do not receive the `LC`, there may be a download problem and you will need to troubleshoot.

```
P 011724 A 040000 B 000000  RW 004507 M 003045 T 100020
VCP>
/DSVCP: %LDS30024
```

VCPMN displays the contents of the slave's memory registers, and DSVCP prompts for a command. In this example, DDL is in file P00003 (and on a FMGR system cartridge) so the user enters %LDS30024, where:

%LDS tells VCP to load a program (%L) via an NS-ARPA/1000 or DS/1000-IV link (DS)

3 the program file name is P00003

00 placeholder value

24 the PSI card with HDLC firmware has a select code 24B.

```
/DSVCP:
LC
```

The slave's VCP sends a Load Complete message. DDL is loaded from file P00003 and ready to be executed.

```
/DSVCP: %E
```

Execute DDL; VCPMN will print its prompt at the master node's system console.

```
DIAGNOSTIC DESIGN LANGUAGE
```

DDL prints a startup message and issues its prompt (>).

```
READY
```

DSVCP prints its prompt (/DSVCP) next to DDL's prompt.

```
>/DSVCP: 10 PRNT 20
>/DSVCP: 20 FMT ("THIS IS A TEST"/)
>/DSVCP: 100 STOP
```

The user types in a simple program.

```
>/DSVCP: LI
10 PRNT 20
20 FMT ("THIS IS A TEST"/)
100 STOP
```

List the program to check for errors.

```
>/DSVCP: RU

THIS IS A TEST
LINE     100 STOP
```

Run the program.

```
>/DSVCP: \E
/DSVCP: END
```

Exit DSVCP.

Programmatic Forced Cold Loads

FCL66 is a utility to programmatically force cold load slave computers over an HDLC link with driver type 66. The slave computer can be an NS-ARPA/1000 node or an E-, F-, or A-Series DS/1000-IV node, and its HDLC link that it uses to connect to the master computer must be select code 24B. This subsection describes how to use DSVCP with an A-Series slave; for more information on forced cold loads to non A-Series nodes, see “Using DSVCP with E- or F-Series.”

Syntax

```
    CALL FCL66 (node,ierr,security[,type][,fnum])
```

Parameters

node — Node number or negative LU number of the link to the system to be force cold loaded. If you specify a node number, it must be described in the NRV as a neighbor and the direct link must be up. If you specify an LU number, and if it is described in the NRV, it must be set in the NRV as a neighbor. For cases where rerouting has taken a node out of the NRV, the LU is still accepted if it is an HDLC driver LU.

ierr — Returned error code (see below).

security — Network management security code.

type — CPU type of remote computer.

0	E- or F-Series
2	A-Series

Default: 0.

fnum — File number (octal). PROGL converts this number to ASCII and uses it to form a `P`*nnnnn* file name. (Not used for E- and F-Series, where the RPL descriptor block bit 0 determines whether file `P00000` or `P00001` is used.) For more information on download files, refer to the “Downloads” subsection.

Range: `0-77777B`

Default: 0.

Error Codes (Octal):

-5	RTE BREAK command was detected.
-4	Error on exit when clearing VCP mode. (A and B registers contain error return. However, if A is between 1 and 15 [inclusive], B is invalid and A is returned from the driver.)
-3	Error return on EXEC call. (A and B registers contain error.)
-2	Parameter missing or error in parameter.
-1	Illegal node number. Not a neighbor nor an HDLC link.
0	No error.
1	Line failure.
2	Time-out.
3	Local busy.
4	Message aborted.
5	Remote busy.
10	Not initialized.
11	Wrong mode.
12	Illegal request.
13	Card failure.
17	Bad interrupt.

Note Even if IERR returns 0, the download may still fail. For example, if the P-file does not exist, or if PROGL is not in the system, the download will fail.

FCL66 verifies that the supplied node number or LU is legal and associated with driver 66. It then sets the link into front-panel-wait mode and sends a break to the remote computer. If the remote computer is an E- or F-Series, front panel wait is cleared and control is returned to the calling program. For A-Series computers, FCL66 waits for a VCP message to come back from the remote computer and then sends a %BDS*nnnnn*0024 RETURN <null> command to the remote machine (*nnnnn* is the ASCII equivalent of fnum). Front panel wait is then cleared and control returned to the calling program. FCL66 waits for a period as long as the link's master time out for the VCP reply from the remote computer or for the link to come back up after VCP mode is entered.

To use FCL66, the switches on the remote interface must be set as described in the previous section, "Required Hardware and Software" in this chapter.

14

Remote I/O Mapping

Overview

Remote I/O mapping is a feature of DS/1000-IV and is also part of the DS/1000-IV Compatible Services of the NS-ARPA/1000 product. Therefore, you can use remote I/O mapping between DS/1000-IV and NS-ARPA/1000-IV nodes. References in this section pertain to both networking products with differences noted. NS-ARPA/1000 runs on RTE-A systems only; DS/1000-IV runs on RTE-A and RTE-6/VM systems.

Remote I/O mapping *maps* or redirects I/O requests destined for an LU on an HP 1000 node to an LU at a remote HP 1000 node. The node at which the mapped LU (or *source LU*) resides is known as the *source node*; I/O requests to the source LU are redirected to the *destination LU* at the *destination node*.

You can use remote I/O mapping to perform the following tasks:

- Establish remote interactive sessions. You can establish a remote interactive session at a remote node. The terminal you are at will seem to be connected directly to the remote computer, with some differences. This remote interactive session access will let you run and interact with programs on that node. Establishing remote interactive sessions is described in detail later in this section.

  In NS-ARPA/1000, TELNET is not supported on a terminal that has been mapped to a remote node with remote I/O mapping.

- Share resources, such as printers, tape drives, and other unit-record devices, throughout the network. The devices cannot be HP-IB devices.

- Accommodate terminal-less nodes. Terminal-less nodes can use remote I/O mapping to send system console output to a terminal on another node. This terminal can receive output from and provide input to a number of terminal-less nodes, allowing it to act as the central control for a network of terminal-less nodes.

- Make programs, interactive and non-interactive, usable at remote nodes without modifying the programs. Output such as error logging from applications programs can be directed to a hard-copy device or perhaps a tape drive at a remote node without program modifications.

The procedures for setting up these uses are discussed later in this section.

Examples

Assume you are at node 1 and there is a magnetic tape unit at LU 8 on node 3. You can establish a map at node 1 that will, transparent to the user, re-direct all output for LU 7 at node 1 to LU 8 at node 3 (see Figure 14-1).

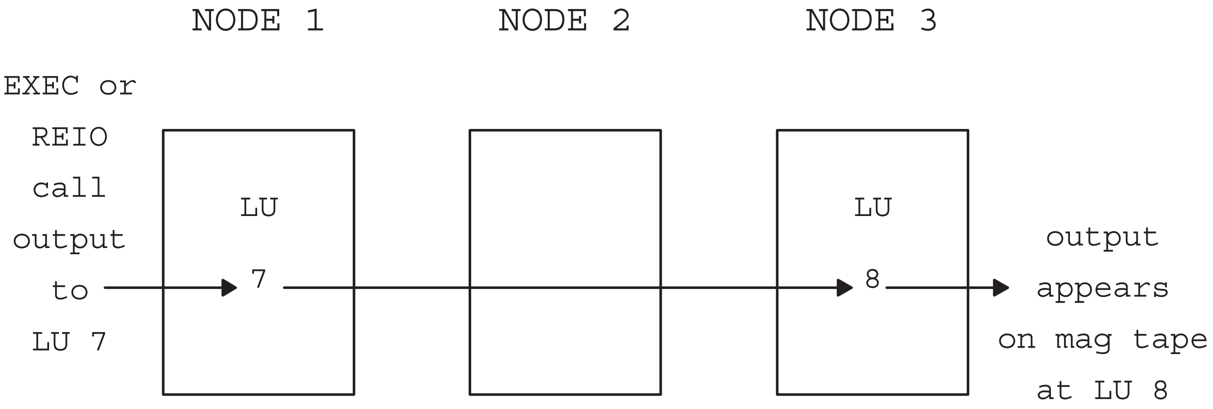


Figure 14-1. Output to LU 7 at Node 1 is sent to LU 8 at Node 3

Then, you can run programs specifying the mapped LU (LU 7 in the above example) as the output LU where, without user program modification, the output is re-directed to be output to an LU at a remote node (LU 8 at node 3 in the above example). In Figure 14-1, there is an intervening node (node 2). Any number of nodes can exist between the two nodes involved.

As another example, assume that you have a terminal-less node (node 1) in a network as shown in Figure 14-2. You would like to have LU 1 available as an interactive output device to act as a system console. In this example, the user has established a remote LU map so that I/O requests destined for LU 1 are re-directed to LU 10 on node 3. All error messages and other output sent to LU 1 on node 1 instead appears on LU 10 at node 3. In a similar manner, any read requests made to LU 1 are re-directed to LU 10 at node 3. The prompt (if any) is displayed and the read is performed at LU 10, node 3 (see Figure 14-2).

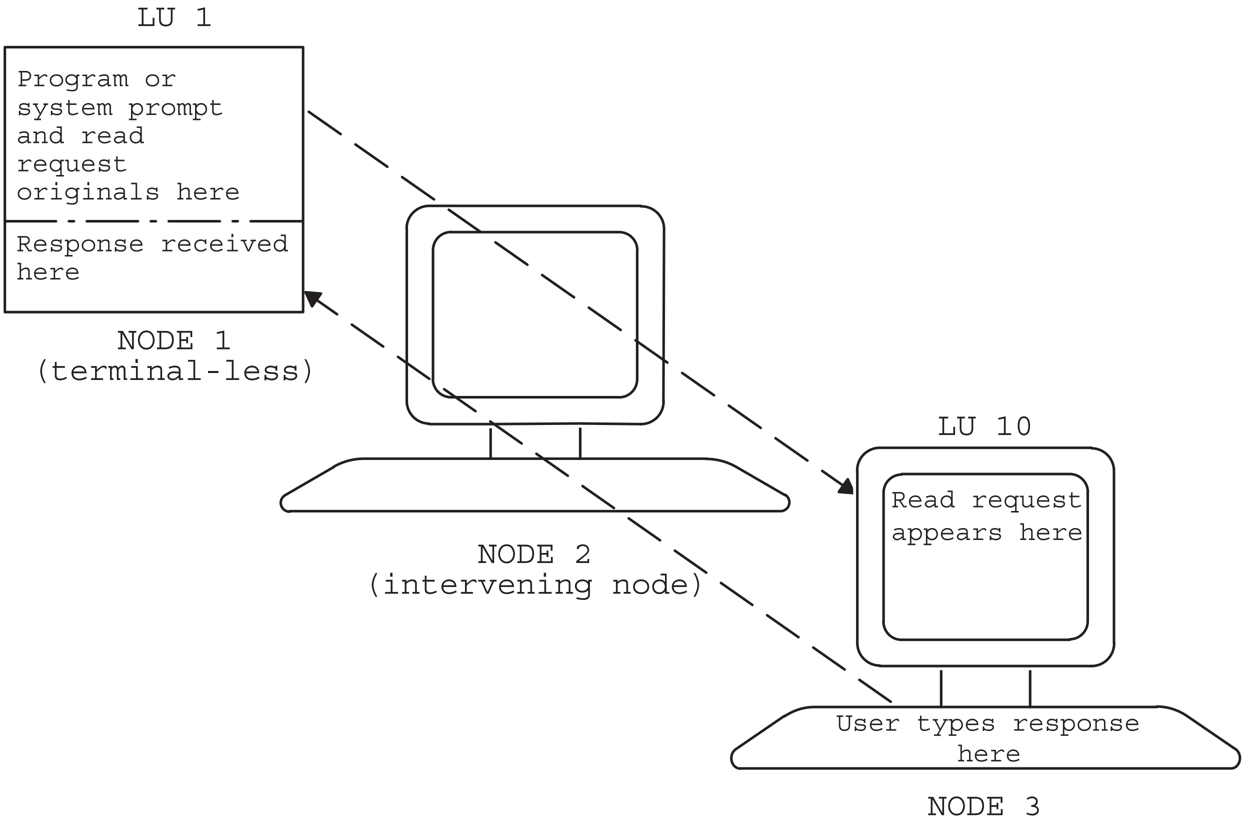

LU 1
Program or system prompt and read request originals here
Response received here
NODE 1 (terminal-less)
NODE 2 (intervening node)
LU 10
Read request appears here
User types response here
NODE 3

Figure 14-2. Read Request on LU 1 Mapped to Node 3 LU 10

In general, whenever an I/O request is made to a mapped LU on one system, an I/O request essentially identical to the request made on the source system is made to the destination LU on the destination system as specified by the map. When the request on the destination system completes, status, driver type, transmission log, and any data read from the destination request are returned in a completion of the source request. Thus the I/O request on the source system is effectively mapped into a request on the destination system.

Note that this mapping is one-way; all transactions are initiated at the source node. That is, writes can be made to LU 1 (appearing on LU 10 at node 3) and read requests can be made to LU 1 (appearing and being satisfied at LU 10 at node 3), but both of these requests are invoked at the source node. Output can appear from node 1 and input can be requested but nothing will be sent to node 1 from node 3 unless explicitly requested by node 1 via a read request.

Remote I/O mapping restrictions are described later in this section.

Mapping I/O To Remote Nodes

A system LU is defined to be mappable if it points to a DVT that has been associated with the remote I/O mapping driver ADV00 in the system generation. Since you can associate LUs to different DVTs with the RTE-A FMGR `LA` command, you can map almost any unit-record type LU you want, subject to remote I/O mapping and FMGR `LA` command restrictions. In RTE-6/VM, the EQT is equivalent to the DVT of RTE-A.

There are three ways to have an LU point to a mappable DVT or EQT:

1. Associate the system LU with the mappable DVT or EQT at system generation time.

2. In RTE-A only, use the FMGR `LA` command to associate the system LU with the mappable DVT. The format of the `LA` command is as follows:

`LA,`*`lu`*`[,`*`dvt`*`]`	sets the *lu* to the specified DVT.
`LA,`*`lu`*	displays the DVT associated with this *lu*.

   In RTE-6/VM, use the system `LU` command to point the system at the mappable EQT.

3. In RTE-6/VM with Session Monitor, the `SL` command can be used to map a session LU to a system LU which already points to a mappable EQT.

Figure 14-3 shows some of the different LU map possibilities. Once again the DVTs of RTE-A are equivalent to EQTs of RTE-6/VM. These are described below:

1. Several LUs can point to the same mappable DVT. See (1) in Figure 14-3.

2. Several mappable DVTs can point to the same destination LU and node. The source LUs need not be at the same node. This can result in interleaved output at the destination LU, but there is an option to identify where each line of output is coming from as it appears on the screen (refer to the IOMAP subsections). See (2) in Figure 14-3.

3. A mappable DVT can point to the bit bucket. A request to an LU pointing to this DVT will be completed with a transmission log of 0 and a status of 0. DVT5 at Source Node B is such an example. See (3) in Figure 14-3.

4. A mappable DVT can point to an LU on the same node. The destination LU cannot be mappable. See (4) in Figure 14-3.

5. A mappable DVT can have no LU pointing to it. This can happen if the LU had been moved elsewhere with the RTE-A FMGR `LA` (or system `LU`) command after the map had been set up. See (5) in Figure 14-3.

If you set up remote I/O mapping interactively from day to day, HP recommends that you use care and develop some tools to manage this resource (refer to “Special Considerations” in this section).

Note Destination LUs cannot be mappable. This prevents DVTs from pointing back to themselves through other mappable DVTs, which would cause infinite loops of I/O requests.

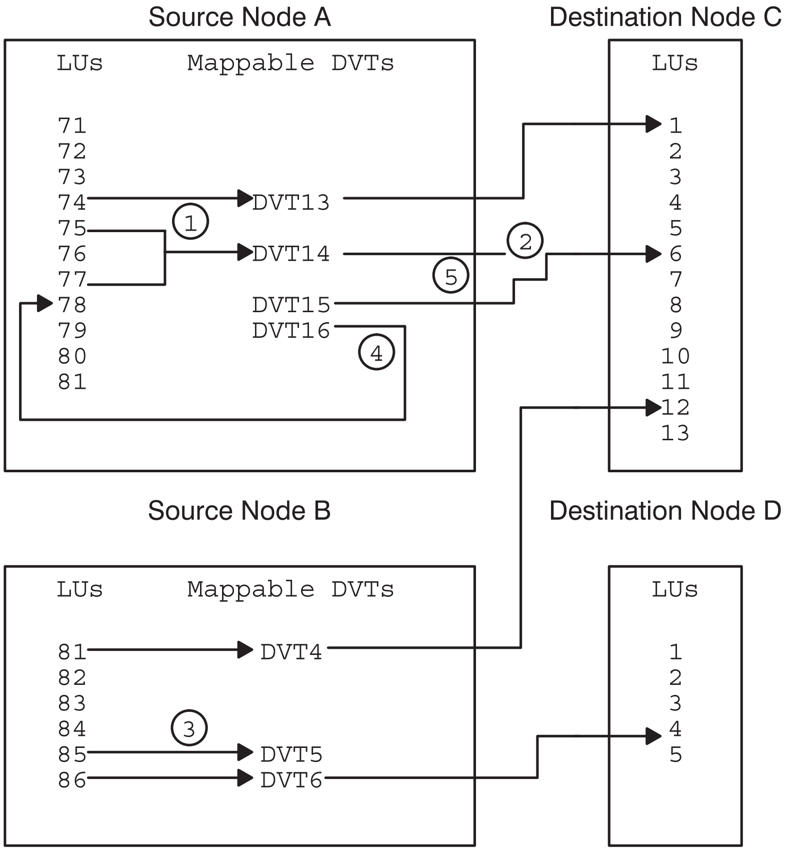


Figure 14-3. Some Remote I/O Mapping Possibilities

Using IOMAP

To set up a map to an LU at a destination node, you execute the program IOMAP at the source node. *You must initialize NS-ARPA/1000 or DS/1000-IV on the source and destination nodes before running IOMAP.*

IOMAP is not interactive. That is, you pass parameters via scheduling parameters and parameters are returned to the parent program (CI or REMAT) via PRTN. IOMAP accepts up to four parameters and returns four parameters. These are described in subsequent sections.

If you run IOMAP from REMAT, REMAT displays the return parameters (in octal). If you run IOMAP from CI, CI places the return parameters into system global variables. You can then display the system global variables with the CI ECHO command.

IOMAP can perform seven functions:

- Initialize I/O Mapping.
- Select the first unused mappable LU and establish a map using that LU.
- Establish or change a map.
- Disable a map.

- Return LU mapping information.
- Return information for a specific mapped LU.
- Return error values.

These functions are summarized in Table 14-1, along with the runstrings. They are explained in detail in the following subsections.

Note There is no protection against re-assigning a map that is in use. You should establish a map by allowing IOMAP to select the first unused mappable LU. You should also clear maps after you have used them.

Table 14-1. Remote I/O Mapping Runstrings

Function	Runstring
Initialize I/O Mapping	`[RU,]IOMAP,`*`mappableLU`*`,-1`
Establish a map using the first unused mappable LU	`[RU,]IOMAP,-1,`*`destLU`*`,`*`destNode`*`,`*`securityCode`*
Establish/change a map	`[RU,]IOMAP,`*`sourceLU`*`,`*`destLU`*`,`*`destNode`*`,`*`securityCode`*
Disable map	`[RU,]IOMAP,`*`sourceLU`*`,0, ,`*`securityCode`*
Obtain information on specific map	`[RU,]IOMAP,`*`sourceLU`*`,-1`
Obtain error values	`[RU,]IOMAP,-2`
Obtain mapping information	`[RU,]IOMAP,`*`sourceLU`*`,-2`

Initialize I/O Mapping

Syntax

```
[RU,]IOMAP,mappableLU,-1
```

Parameters

mappableLU Mappable LU, which is a system LU associated with a mappable DVT in RTE-A or mappable EQT in RTE-6/VM.

Return Parameters

First Parameter IOMAP returns the specified *mappableLU* if it is mappable; if *mappableLU* is not mappable, this will be the first mappable LU greater than *mappableLU*.

Otherwise, it is one of the following negative error codes:

- -2 (177776B) if the *sourceLU* is not a mappable LU and there are no mappable LUs greater than *sourceLU*
- -3 (177775B) if remote I/O mapping is not set up correctly

Discussion

Before establishing any maps, you must initialize I/O Mapping. You may want to initialize IOMAP in the system WELCOME file, after initializing NS-ARPA/1000 or DS/1000-IV. This does *not* establish a map for *mappableLU*, which is still available.

`CI>RU,IOMAP,72,-1`	Initialize I/O Mapping, specifying LU 72. Note that this does not establish a map for LU 72; LU 72 is still available.

`CI>ECHO,$RETURN1`
`   72`
	Display return parameter.
	`72` indicates that I/O Mapping is initialized.

Establish a Map Using First Unused Mappable LU

Syntax

```
[RU,]IOMAP,-1,destinationLU,destinationNode,securityCode
```

Parameters

destinationLU Destination system LU of map.

Options:

Set bit 15 to enable the header flag. Remote I/O mapping will write a message of the following form to the destination LU preceding each record sent to that LU:

```
MESSAGE FROM NODE # nnnnn PRGM ppppp AT DAY dddd, hh :mm :ss
```

Where: *nnnnn* is the source node (node where the message originated); *ppppp* is the name of the program sending the message (or `SYS I/O` if the I/O request is from the system or class buffered); and *ddddd hh mm ss* is the day and time that the request was made at the source node.

Set bit 14 to enable the prompt flag and set the read request timeout to 5 minutes. Remote I/O mapping will write a prompt of the following form to the destination LU preceding each read request mapped from the source LU:

(*nnnnn*)

Where: *nnnnn* is the source node. Refer to the discussion below for more information.

Set bit 13 with bit 14 to change the read request timeout to 5 minutes but suppress the prompt flag.

destinationNode Destination node number of the map (the node of the destination LU).

securityCode Network security code.

Return Parameters

First Parameter

The *sourceLU* is returned if the map was successfully established, otherwise one of the following negative error codes is returned:

- -1 (177777B) if *securityCode* was incorrect
- -2 (177776B) if *sourceLU* is not a mappable LU
- -3 (177775B) if remote I/O mapping is not set up correctly or if Remote I/O Mapping has not been initialized
- -4 (177774B) if *destinationLU* is mappable
- -5 (177773B) if *destinationNode* is not known to the local node
- -6 (177772B) if *destinationLU* is invalid or if attempts to obtain information about the destination LU failed
- -7 (177771B) if LUMAP is not present and active
- -8 (177770B) if LUQUE is not present and active

Second Parameter

destinationLU

Third Parameter

destinationNode

Fourth Parameter

If IOMAP did not establish the map, it returns the specified *securityCode*.

If IOMAP successfully established the map, it returns one of the following values:

- 0 (00000B) normal completion
- -1 (177777B) if *destinationLU* has either an associated infinite device timeout or a timeout larger than the user requested master timeout
- -2 (177776B) if *destinationLU* is associated with the bit bucket
- -3 (177775B) if *destinationLU* is not a unit-record device

Discussion

Use this runstring when you want to use a mappable LU that is not already mapped (for example, when sharing a pool of these LUs among various users). IOMAP will search through the mappable LUs until it finds one that is not mapped and then use that LU as the source LU.

Prompt Flag

Setting the prompt flag bit (bit 14) is useful when several LUs on several nodes are mapped to the same LU on one node. Users can then distinguish between the input and output requests from different nodes. Setting bit 14 also changes the read request timeout so that you have 5 minutes to respond to read requests at this LU. Refer to “Timeouts” in this section for more information.

Setting bit 13 with bit 14 sets the read request timeout to 5 minutes but suppresses the prompt flag.

Example

```
CI>RU,IOMAP,-1,60010B,3,NM
```

Map first available mappable LU to LU 8 at node 3 with the prompt flag option (bit 14). The network security code is NM.

```
CI>ECHO,$RETURN1,$RETURN2,$RETURN3,$RETURN4
```

Display return parameters.

```
   83,8,3,-1
```

`83` is the source LU and indicates a successful mapping.
`8` is the destination LU.
`3` is the destination node.
`-1` indicates that the destination LU (LU 10) has a timeout that is greater than the master timeout

Establish or Change a Map

Syntax

```
[RU,]IOMAP,sourceLU,destinationLU,destinationNode,securityCode
```

Parameters

sourceLU Mappable LU on source system.

destinationLU Destination system LU of map. The *destinationLU* must be associated with a unit-record device (such as a terminal, printer, or mag tape unit).

Options:

Set bit 15 to enable the header flag. Remote I/O mapping will write a message of the following form to the destination LU preceding each record sent to that LU:

```
MESSAGE FROM NODE # nnnnn PRGM ppppp AT DAY dddd, hh :mm :ss
```

Where: *nnnnn* is the source node (node where the message originated); *ppppp* is the name of the program sending the message (or `SYS I/O` if the I/O request is from the system or class buffered); and *ddddd hh mm ss* is the day and time that the request was made at the source node.

Set bit 14 to enable the prompt flag and set the read request timeout to 5 minutes. Remote I/O mapping will write a prompt of the following form to the destination LU preceding each read request mapped from the source LU:

(*nnnnn*)

Where: *nnnnn* is the source node. Refer to the discussion below for more information.

Set bit 13 with bit 14 to change the read request timeout to 5 minutes but suppress the prompt flag.

destinationNode Destination node number of the map (the node of the destination LU).

securityCode Network security code.

Return Parameters

First Parameter

The *sourceLU* is returned if the map was successfully established, otherwise one of the following negative error codes is returned:

- -1 (177777B) if *securityCode* was incorrect
- -2 (177776B) if *sourceLU* is not a mappable LU
- -3 (177775B) if remote I/O mapping is not set up correctly
- -4 (177774B) if *destinationLU* is mappable
- -5 (177773B) if *destinationNode* is not known to the local node
- -6 (177772B) if *destinationLU* is invalid or if attempts to obtain information about the destination LU failed
- -7 (177771B) if LUMAP is not present and active
- -8 (177770B) if LUQUE is not present and active

Second Parameter

destinationLU

Third Parameter

destinationNode

Fourth Parameter

If IOMAP did not establish the map, it returns the specified *securityCode*.

If IOMAP successfully established the map, it returns one of the following values:

- 0 (000000B) normal completion
- -1 (177777B) if *destinationLU* has either an associated infinite device timeout or a timeout larger than the user requested master timeout
- -2 (177776B) if *destinationLU* is associated with the bit bucket
- -3 (177775B) if *destinationLU* is not a unit-record device

Discussion

To change a map, run IOMAP as you would to establish a map. Specify the source LU and the new destination LU (with or without header and prompt flags) and node.

Caution There is no protection against re-assigning a map that is in use. You should clear maps after using them, and allow IOMAP to select the first unused LU to establish a map (described in the previous subsection).

Prompt Flag

Setting bit 14 (prompt flag) is useful when several LUs on several nodes are mapped to the same LU on one node. Users can then distinguish between input and output requests from different nodes. Setting bit 14 also changes the read request timeout so that you have 5 minutes to respond to read requests at this LU. Refer to “Timeouts” in this section for more information.

Setting bit 13 with bit 14 sets the read request timeout to 5 minutes but suppresses the prompt flag.

Example

`CI>RU,IOMAP,72,100015B,50,NM`	Map LU 72 to LU 13 at node 50 with the header flag option (bit 15). The network security code is NM.

`CI>ECHO,$RETURN1`
`   72`
	Display the first return parameter.
	72 indicates a successful mapping.

Disable Map

Syntax

```
[RU,]IOMAP,sourceLU,0,0,securityCode
```

Parameters

sourceLU	System LU of map to be disabled.
securityCode	Network security code.

Return Parameters

First Parameter — The *sourceLU* is returned if the map was successfully disabled, otherwise one of the following negative error codes is returned:

- -1 (177777B) if *securityCode* was incorrect
- -2 (177776B) if *sourceLU* is not a mappable LU
- -3 (177775B) if remote I/O mapping is not set up correctly
- -7 (177771B) if LUMAP is not present and active
- -8 (177770B) if LUQUE is not present and active

Second Parameter	0 (the destination LU is the bit bucket)
Third Parameter	0
Fourth Parameter	specified *securityCode*

Discussion

Disabling a map makes the specified LU available and a candidate for selection when IOMAP is looking for an unused mappable LU.

All maps to and from a node must be disabled before shutting down DS/1000-IV.

Example

`CI>RU,IOMAP,88,0,,NM`	Disable the map for LU 88. The network security code is NM.

`CI>ECHO,$RETURN1`
`88`
	Display the first return parameter.
	`88` indicates that the source LU, 88 was successfully disabled.

Return LU Mapping Information

Syntax

```
[RU,]IOMAP,sourceLU,-1
```

Parameter

sourceLU The mappable LU for which you want IOMAP to display information. If you specify a non-mappable LU, IOMAP will display map information for the first mappable LU greater than *sourceLU*.

Return Parameters

First Parameter The *sourceLU*.

If *sourceLU* is not mappable, this will be the first mappable LU greater than *sourceLU*.

Otherwise, it is one of the following negative error codes:

- -2 (177776B) if the *sourceLU* is not a mappable LU and there are no mappable LUs greater than *sourceLU*
- -3 (177775B) if remote I/O mapping is not set up correctly

Second Parameter destination LU of map for the LU returned in the first parameter.

Third Parameter destination node number of map for the LU returned in the first parameter.

Fourth Parameter The type of remote operating system (`$OPSYS` value, as documented in the RTE-A manual set).

Discussion

You can use this runstring to determine which LUs in a system are mappable or to determine what map, if any, has been set up for a specific mappable LU.

The return parameters show the current map for *sourceLU*.

Example

```
CI>RU,IOMAP,68,-1
```

Return mapping information for LU 68.
If LU 68 is not mappable, IOMAP will return mapping information for the first mappable LU greater than 68.

```
CI>ECHO,$RETURN1,$RETURN2,$RETURN3,$RETURN4
```

Display return parameters.

```
    74,8,101,-53
```

Indicates that the first mappable LU greater than 68 is 74, and that LU 74 is mapped to LU 8 at node 101, which is an RTE-A node.

Return Mapping Information for a Specific LU

Syntax

```
[RU,]IOMAP,sourceLU,-2
```

Parameters

sourceLU The mappable LU for which you want IOMAP to display information.

Return Parameters

First Parameter *sourceLU*, or the following negative error code:

- -3 if I/O mapping is not initialized correctly.

Second Parameter destination node, or the following negative error code:

- -1 (177777B) if *sourceLU* is not mappable

Third Parameter destination LU

If the first return parameter is not a mappable LU, IOMAP returns one of the following negative error codes:

- -1 if LU is an invalid LU number
- -2 if LU is associated with the bit bucket
- negative of the timeout of the LU's DVT otherwise

Fourth Parameter The type of remote operating system (`$OPSYS` value, as documented in the RTE-A manual set).

Discussion

This runstring returns mapping information for the specified LU.

Example

```
CI>RU,IOMAP,77,-2              Return mapping information for LU 77.
CI>ECHO,$RETURN1,$RETURN2,$RETURN3,$RETURN4
                               Display return parameters.
    77,6,50,-53                Indicates that LU 77 is mapped to LU 6 at node 50, which
                               is an RTE-A node.
```

Return Error Values

Syntax

```
[RU,]IOMAP,-2
```

Return Parameters

First Parameter	first word (two characters) of the ASCII four-character error code (for example, “DS” of DS04). For serious errors, this is a numeric error code, and the second and third parameters (below) are meaningless.
Second Parameter	second word (two characters) of the ASCII four-character error code.
Third Parameter	number of node reporting the error.

Discussion

The return parameters contain the values at the location #LMPE, #LMPE+1 and #LMPE+2. If remote I/O mapping detects an error, it stores the error code in the doubleword #LMPE (at the source node) and the reporting node number in #LMPE+2. When an error occurs or if no map is set up for a called mappable DVT, the request is completed with a transmission log of zero and a status of zero.

The errors at #LMPE are usually four character ASCII error codes, but if a serious problem is detected, a numeric code (an integer from 0 to 12) is set at #LMPE, in which case the ASCII codes at #LMPE+1 and #LMPE+2 are meaningless. Refer to Table 14-2 for the #LMPE numeric error codes.

Note that #LMPE is unchanged unless an error occurs. That is, it always contains the last error encountered by the remote I/O mapping driver.

#LMPE error values are intended for use by the Network Manager for troubleshooting information if remote I/O mapping is not working properly.

Example

```
CI>RU,IOMAP,-2     Return error information.
```

Table 14-2. #LMPE Numeric Error Codes

Code	Meaning
0	No error.
1	Reserved for future use.
2	Reserved for future use.
3	Reserved for future use.
4	DS/1000-IV or NS-ARPA/1000 quiescent resource number is corrupt.
5	Source node is quiescent.
6	Class number set up for LUMAP has been corrupted.
7	Reserved for future use.
8	Program LUMAP is not present in system.
9	Destination node number is not in the NRV.
10	DS/1000-IV or NS-ARPA/1000 is not initialized.
11	Error on class number allocation.
12	No class numbers are available.

Remote Interactive Sessions

Remote I/O mapping allows you to gain access to the remote command interpreters (CI or FMGR). This “virtual terminal” capability allows you to establish a session at a remote node and interact with the command interpreter as if the terminal were connected directly to the remote node.

To set up a remote interactive session, you must perform the following tasks:

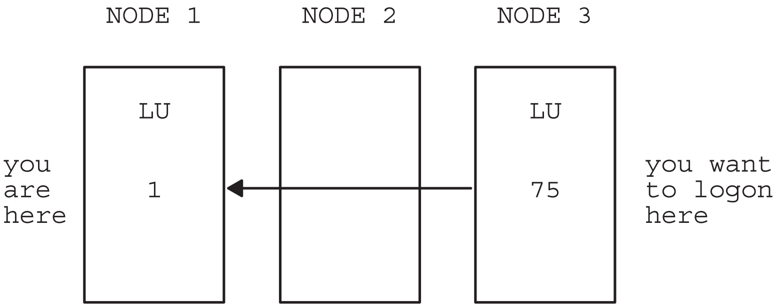
- Run IOMAP from REMAT to establish an LU map at the remote node.
- Run SYSAT to get the remote system’s attention and logon prompt.
- Logon to the remote node.

These steps are described in the following subsections.

Establishing the Map

Use REMAT to switch to the node at which you want to establish an interactive session. Then, run IOMAP at the remote node, mapping an LU at that node to your terminal’s LU at your node.

For example, assume you are at node 1 terminal LU 1 and you want to establish an interactive session at node 3. Using REMAT, you switch to node 3 and run IOMAP, mapping an LU to our terminal LU on the local node. Refer to Figure 14-4.



```
CI>RU REMAT                  Run REMAT at node 1

$SW,3,,DS                    Switch to node 3 (Network User's
                             security code = DS)


#RW,IOMAP,75,40001B,1,NM
    000113 040001B 000001 047115 00000
                                          Map LU 75 to LU 1
                                          at node 1, setting
                                          the prompt flag
                                          (Network security
                                          code = NM)

                               Octal ASCII coded "NM"

                         destination node of map

             destination LU (in octal)
             with prompt option specified

    source (mappable) LU
    (octal representation of 75)


      NOTE:  Return parameters displayed as octal values.

#EX
CI>
```

Figure 14-4. Establish LU Map at Remote Node

Get the Remote System's Attention

The next step is to get the remote system's attention. To do this, run the program SYSAT, described below.

SYSAT—System Attention Program

You can use SYSAT to get system attention or to set the break bit of a program on a remote node. This is useful for logging on to remote systems, or for a remote node that has no terminals of its own but uses Remote I/O mapping to communicate. In gaining system attention on a particular LU, SYSAT has the effect of someone striking a key on a terminal to get a logon (or break mode) prompt.

To get system attention on a remote node, enter:

```
    RU,SYSAT,attentionLU,remoteNode
```

This causes the remote node to give its breakmode prompt or the logon prompt on the `attentionLU`. The `attentionLU` on the remote node must be a mappable LU. If the remote system is multiuser, the system prompt program is scheduled, and the logon prompt is displayed. If the remote system is not multiuser, the `RTE:` prompt is displayed. In RTE-6/VM systems, the system prompt is given if the LU is 1. Otherwise, the program associated with remote I/O mapping, if any (specified in the interrupt table), is scheduled and passed the DVT address in RTE-A (EQT address in RTE-6/VM). At least one map must have been set up at the specified node for this request to work. Also, IOMAP must be present and active at the remote node or SYSAT returns a `NO PROGRAM ID` error (`NO SUCH PROGRAM` if the remote node is an RTE-6/VM system).

To set the break flag of a program in a remote node, enter:

```
    RU,SYSAT,programName,remoteNode
```

Logon

If you are running under a command interpreter at the local node, then you will want to keep the local command interpreter from interfering with the remote system. One way to do this is to suspend (`SS`) the local command interpreter. To restart CI, enter the system command `GO`. Refer to the *RTE-A User's Manual*, part number 92077-90002, for more information on these commands. The procedure for suspending the command interpreter, obtaining system attention via SYSAT, and logging on is illustrated in Figure 14-5. In this example, the command interpreter at the local and remote nodes is CI.

You must enter the password, if any, along with the account name (on the same line) in response to the logon prompt.

```
  CI>SS,CI
  <CR>
  CM>RU,SYSAT,75,3
  MESSAGE DELIVERED

  SYSTEM    3 LOGON: (     3) USER/PASSWORD

  CI>(     3)      {now enter commands as if your terminal
     .              were directly connected to node 1}
     .
     .
  CI>(     3)EX
  DONE

RETURN
  CM>GO,CI
  CI>
```

Figure 14-5. Logon to Remote Node (User Input Underlined)

In RTE-6/VM, one way to keep your local FMGR from interfering with your remote session is to simply `OF` it. An example procedure is shown in Figure 14-6.

```
:OF,FMG01

FMG01 ABORTED
RETURN
S=01 COMMAND ? RU,SYSAT,75,3
MESSAGE DELIVERED

SYSTEM   3 LOGON: (    3) USER.GENERAL/PASSWORD

SESSION 75 ON. 4:14 PM MON.,  18 AUG., 1980
PREVIOUS TOTAL SESSION TIME: 05 HRS., 04 MON., 58 SEC.

:(    3)       {now enter commands as if your terminal
   .            were directly connected to node 1}
   .
   .
:(    3)EX,SP
 $END FMGR
FMG75 REMOVED

SESSION  75  OFF 4:14 PM  MON., 18 AUG., 1980
CONNECT TIME:             00 HRS., 00 MIN., 34 SEC.
CPU USAGE:                00 HRS., 00 MIN., 00 SEC., 350 MS.
CUMULATIVE CONNECT TIME: 05 HRS., 05 MIN., 32 SEC.
END OF SESSION
RETURN
S=01 COMMAND ?RS
:
```

Figure 14-6. Other RTE Systems: Logon to Remote Node

Obtaining a CM or Breakmode Prompt

While logged on to a remote node, there may be times where you want the CM's attention (or breakmode prompt). Typically, the CM (or breakmode) prompt you obtain will be that of the local system. To get the remote node's CM, obtain the local node's CM prompt and run SYSAT, specifying the LU mapped to this node as the attention LU (or, if running a program, you can also specify the program name). SYSAT will cause the remote node's CM (or breakmode) prompt to appear.

Remote I/O Mapping Considerations

This section summarizes remote I/O mapping considerations.

Double-Buffered Calls (Z-BIT)

The double-buffered Z-bit option used in EXEC reads and writes cannot be used when performing an EXEC read or write to a mapped LU. It is for this reason that communication to some HP-IB devices via a mappable LU is *not* supported. See also “HP-IB and Disk LUs Not Mappable” later in this section.

Double-buffered reads from RTE-A are always translated into an interactive write/read.

If the NB bit (nonbuffered operation) is used to a mapped LU, it is ignored.

Data Buffers Must Not Exceed 1024 Words

Data buffers specified in I/O requests must not exceed 1024 words. This should not be a problem since remote I/O mapping is intended for use with unit record devices.

Timeouts

After logging on to a remote node, the remote command interpreter issues a read request to the remote mapped LU, which is re-directed to the local terminal LU via a DEXEC call. This DEXEC read request is subject to the NS-ARPA/1000 or DS/1000-IV master request timeout value set at the remote node (the default is 45 seconds). If you do not respond to the command interpreter request within the master timeout period, the following events occur:

At the system where you have the remote session:

> A zero length record completion is recorded, causing the command interpreter to re-prompt and send another read request. In RTE-6/VM, FMGR will write `WAITING FOR INPUT` to the mapped LU and send another read request. These are translated into DEXEC calls, mapped to the destination node, and queued behind the original read request still pending on the terminal.

At the local node where the terminal is physically attached:

> While the read request times out at the remote node, it remains pending on the terminal at the local node.
>
> When you respond to the read request, the line you typed is flushed because that DEXEC reply is no longer expected at the remote node.
>
> The remote command interpreter re-prompts you, as if you typed carriage return and nothing else. In RTE-6/VM, the FMGR message `WAITING FOR INPUT` appears on your terminal with another prompt as if you typed carriage return and nothing else.

If enough time elapses, several of these NS-ARPA/1000 or DS/1000-IV master timeouts may occur. The result would be several of the command interpreter's read requests completing with zero length records. If enough read requests timeout in this way while under the command interpreter, you will be logged off of the remote session.

Prompt Flag Timeout

Specifying the "prompt" flag (bit 14 on the destination LU) when establishing a map with IOMAP will cause the command interpreter's DEXEC read requests to be performed with the interactive write/read option, allowing the request 5 minutes to complete. The NS-ARPA/1000 or DS/1000-IV master timeout (no prompt flag) and the interactive write/read timeout (prompt flag) apply to all read requests made to a mapped LU, not just the command interpreter's. Note that your local terminal timeout will function in the way you would normally expect it to, even though the session is a remote session.

In RTE-A if the I/O mapping timeout is shorter than the command interpreter timeout, you must hit multiple carriage returns to obtain the command interpreter's attention again. These multiple carriage returns also cause slave TCB (Transaction Control Block) timeout errors. If the command interpreter's timeout is less than the I/O mapping timeout, then the command interpreter will complete a read request with a timeout, and another prompt appears. No error message is displayed. This situation is not a problem, but may appear confusing as the prompt appears without being invoked by the user.

EDIT/1000

When running a screen mode editor such as EDIT/1000, you should set bits 13 and 14 of the destination LU parameter. This sets the read request timeout to 5 minutes and suppresses the prompt flag so that the prompt flag option will not affect screen edits.

LUs Mapped to Terminals Use EQT Subchannel 0 (RTE-6/VM)

In RTE-6/VM although any subchannel on a mappable EQT can be used by an LU, it is strongly recommended that each terminal LU (or interactive device) use EQT subchannel zero. This is because many programs check the subchannel of the LU that they are passed or call the RTE routine `IFTTY`. A non-zero subchannel is interpreted by these programs to mean that the device is not interactive.

Also, system prompts always go to the EQT without a subchannel specification. Thus the system prompt will go to subchannel zero. There is no distinction made between subchannels here. Running SYSAT and specifying an LU causes the prompt to be sent to the device associated with subchannel zero regardless of the subchannel actually associated with the specified LU.

LUs Mapped to Terminals Should Be Less Than 100

When logging on to a remote session via a mapped LU, the mapped LU must be less than 100. This prevents duplication of cloned program names.

HP-IB and Disk LUs Not Mappable

You cannot map mappable LUs to LUs that are connected to disk LUs or HP-IB devices. Disks are not unit record devices and can easily exceed the 1024 word buffer size limit.

HP-IB libraries use double-buffered (Z-bit) EXEC calls for normal communication. It is for this reason that communication to HP-IB devices via mappable LUs is not supported.

EXEC 3 (Control Request)

Control requests over I/O mapping do not work if there are any optional parameters specified.

EXEC 13 (I/O Status)

EXEC 13 requests obtain information (status and device type) about the device associated with a specified LU. You can call EXEC 13 for local (mapped) LU , and remote I/O mapping will return the appropriate information for the DVT at the node of the mapped LU. Since several different devices (LUs) may be mapped to the same DVT, performing an EXEC 13 will give you the status word (DVT word 5) for the most recent operation, regardless of the LU. This means that if a DVT is used to map two LUs and you request status of one of these LUs, you may get the status of the other LU if it was the last one to perform an operation through that DVT.

In RTE-6/VM, EQT is the equivalent to DVT of RTE-A.

Note that DVT word 4 (or EQT word 5) is not provided. Any value returned for this word is meaningless.

Sharing Devices

LUs at remote nodes are not locked when data is mapped to them from local nodes. You must coordinate device use between all users to avoid situations such as interleaved output on remote lineprinters.

Remote I/O Mapping Reserved DVT/LU

A reserved DVT or LU is specified at system generation time for remote I/O mapping software. Do not establish user maps with the LU specified by this DVT.

Accessing the HP 3000

When you use remote I/O mapping to establish a remote logon, you have access to all the programs at that system. This includes the program RMOTE if the node you are logged on to is directly connected to an HP 3000 via a Bisync or X.25 link.

Thus, you can run RMOTE, establish an HP 3000/MPE session, and have a “virtual terminal” connection to the HP 3000 from any terminal in the network via remote I/O mapping. However, you must remember that you are actually logged on to the HP 1000 that is the neighbor node to the HP 3000. This means that file transfers can be performed only between the HP 3000 and its neighboring HP 1000 node. To transfer a file from your local node to the HP 3000 (or vice versa), first use REMAT to transfer the file to the HP 3000’s neighboring HP 1000. Then use remote I/O mapping to logon to the remote HP 1000, run RMOTE and perform the file transfer to the HP 3000.

Remote I/O mapping cannot be used to map an LU on an HP 1000 to an HP 3000.

Note HP 3000 applications that reset terminal straps or make use of terminal block-mode capabilities may not function properly when run from RMOTE due to differences between HP 3000 and HP 1000 I/O systems, drivers, and interface cards.

TELNET (NS-ARPA/1000 Only)

TELNET is not supported on a terminal that has been mapped with remote I/O mapping to a remote HP 1000 node.

Error Processing

Programs performing input, output, and control requests to a mapped LU will have driver status returned in the A-register and the transmission log returned in the B-register upon successful completion. Unsuccessful completions may result from several causes:

1. An NS-ARPA/1000 or DS/1000-IV error. The A-register will return with bit 5 set (timeout) and the B-register will be set to zero. The actual error will be placed in #LMPE (you can examine it with IOMAP).

2. An EXEC error. If the EXEC error is detected at the local system (e.g., IO01—not enough parameters) the request is handled as it would be for any other EXEC request with that error.

   If the error is detected at the remote node, the A-register on return will have bit 5 set, and the B-register will be zero. #LMPE will contain the error code. An error encountered at the remote node will cause return to the normal completion return point when the no-abort bit is set.

3. A driver error at the remote system. If the remote driver detects a condition which causes the remote system to print a message on its system console, then upon return to the program:

   - the A-register will have bit 5 set
   - the B-register (transmission log) will be zero
   - #LMPE will contain IO07

If no message is printed at the remote system console, the driver status is returned in the A-register as a normal completion and the transmission log will be set as in a normal completion.

A

Configuration Worksheets

This appendix contains the following items:

- blank network configuration worksheet
- blank routing worksheet
- examples of network configuration worksheets from the following example files:

  ```
  /NS1000/EXAMPLES/EX_802_WORKSHEET.TXT
  /NS1000/EXAMPLES/EX_RTR_WORKSHEET.TXT
  ```

- examples of NSINIT answer files for initialization:

  ```
  /NS1000/EXAMPLES/NODE1_802.NSIN
  /NS1000/EXAMPLES/NODE2_802.NSIN
  /NS1000/EXAMPLES/NODE1_RTR.NSIN
  /NS1000/EXAMPLES/NODE2_RTR.NSIN
  ```

- example of an NRINIT answer file for nodal registry initialization:

  ```
  /NS1000/EXAMPLES/ALL_NODES.NRIN
  ```

For more information and examples, refer to Section 3, “Network Configuration Planning.”

Blank Network Configuration Worksheet

The following pages contain a blank network configuration worksheet, from the file /NS1000/EXAMPLES/NS_WORKSHEET.TXT.

```
                     NS_WORKSHEET.TXT 91790-17045 REV.6100 <930803.1134>
                   NS/1000  NETWORK  WORKSHEET
                  ----------------------------


Note: The same number of asterisks (*, **) indicates related information.
      () indicate default values.  /D indicates default value taken.
      For LAN/1000 links, user LU = even value = read LU - 1
      For all other links, transmit LU = receive LU - 1.
      Node Number is equivalent to Router/1000 address.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
NODAL INFORMATION
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

1.Node Name
-------------------|-------------------|-------------------|------------------
2.IP Addresses
-------------------|-------------------|-------------------|------------------
3.Router/1000 Address
-------------------|-------------------|-------------------|------------------
4.Files


NSINIT (NSOUTxx,
xx=1 to 99)
-------------------|-------------------|-------------------|------------------
NRINIT
(LU 1)
-------------------|-------------------|-------------------|------------------
NSTRC
(NS_TRACE.TRC)
-------------------|-------------------|-------------------|------------------
FMTRC
(NS_TRACE.FMT)
-------------------|-------------------|-------------------|------------------
EVMON
(/SYSTEM/NS_EVENT.LOG)
-------------------|-------------------|-------------------|------------------
5. Link Information
   Valid Link Interface | Communication link combinations:
             LAN        |      LAN
             RTR        |      HDLC, X.25 (RTE-RTE)
             none       |      X.25 (RTE-MPE) or Bisync
LAN Station addresses are in hexadecimal.

Communication link
-------------------|-------------------|-------------------|------------------
Link Interface type
```

```
-------------------|-------------------|-------------------|------------------
  user or     read or
transmit LU, receive LU
-------------------|-------------------|-------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|-------------------|-------------------|------------------

Communication link
-------------------|-------------------|-------------------|------------------
Link Interface type
-------------------|-------------------|-------------------|------------------

  user or     read or
transmit LU, receive LU
-------------------|-------------------|-------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|-------------------|-------------------|------------------

Communication link
-------------------|-------------------|-------------------|------------------
Link Interface type
-------------------|-------------------|-------------------|------------------

  user or     read or
transmit LU, receive LU
-------------------|-------------------|-------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|-------------------|-------------------|------------------

** Pool LUs for
X.25 MPE links
-------------------|-------------------|-------------------|------------------
6.Network Membership

*** No. of networks to which
node is directly connected
-------------------|-------------------|-------------------|------------------

7.Services

NFT (Y)
-------------------|-------------------|-------------------|------------------
*DS/1000-IV
Compatible Serv (N)
-------------------|-------------------|-------------------|------------------
   1000 - 1000 (Y)
   ----------------|-------------------|-------------------|------------------
 **1000 - 3000 (N)
   ----------------|-------------------|-------------------|------------------
   All monitors (Y)
   VCPMN, TRFAS, RDBAM,
   RFAM, PROGL, PTOPM,
   OPERM, EXECW, EXECM,
   DLIST, CNSLM
-------------------|-------------------|-------------------|------------------

8.Probe Information

Proxy Nodal Registry
Server (N)
-------------------|-------------------|-------------------|------------------
Target Address
(09-00-09-00-00-01)
-------------------|-------------------|-------------------|------------------
Proxy  Address
(09-00-09-00-00-02)
-------------------|-------------------|-------------------|------------------

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
ROUTING INFORMATION
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

9.***Directly Connected Network (DCN) Information (for RTR and 802 LIs)

IP Address
-------------------|-------------------|-------------------|------------------
RTR or 802
-------------------|-------------------|-------------------|------------------
segsize (8000 if RTR,
         1500 if 802)
-------------------|-------------------|-------------------|------------------
if 802, user LU
-------------------|-------------------|-------------------|------------------
if 802, Enable/
Not Enable (E)
-------------------|-------------------|-------------------|------------------

10. Gateway Table (GT) Information

Dest IP net
-------------------|-------------------|-------------------|------------------
Gateway IP
-------------------|-------------------|-------------------|------------------
Hops (100)
-------------------|-------------------|-------------------|------------------

11. * NRV Entries for remote nodes
    Make an entry for each remote node that:
      a)supports DS/1000-IV Compatible Services (RTE-RTE)
    or
      b)is connected via a RTR link interface.

    Where:
      Node Number is equivalent to Router/1000 address.
      Valid choices for Neighbor node are:  N or blank
      Valid choices for Message Accounting are:  MA or blank
```

```
Node Number
-------------------|-------------------|-------------------|------------------
IP address
-------------------|-------------------|-------------------|------------------
If no rerouting, LU
-------------------|-------------------|-------------------|------------------
LU timeout (45 sec)
-------------------|-------------------|-------------------|------------------
Level (1)
-------------------|-------------------|-------------------|------------------
Neighbor node
-------------------|-------------------|-------------------|------------------
Message Accounting (no MA)
-------------------|-------------------|-------------------|------------------
MA timeout (3)
-------------------|-------------------|-------------------|------------------

Node Number
-------------------|-------------------|-------------------|------------------
IP address
-------------------|-------------------|-------------------|------------------
If no rerouting, LU
-------------------|-------------------|-------------------|------------------
LU timeout (45 sec)
-------------------|-------------------|-------------------|------------------
Level (1)
-------------------|-------------------|-------------------|------------------
Neighbor node
-------------------|-------------------|-------------------|------------------
Message Accounting (no MA)
-------------------|-------------------|-------------------|------------------
MA timeout (3)
-------------------|-------------------|-------------------|------------------

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
SECURITY CODES
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

12. Network Security Code
-------------------|-------------------|-------------------|------------------
User's Security Code
-------------------|-------------------|-------------------|------------------
13.**Bisync Dial-Up
Local ID Sequence
-------------------|-------------------|-------------------|------------------
Remote ID Sequence
-------------------|-------------------|-------------------|------------------
```

Blank Routing Worksheet

A blank Routing Worksheet is printed on the next page.

Node Name IP Address(es) Router/1000 Address			
DCN Information *local IP addr*, RTR or LAN, *segsize*, If LAN: *link lu*, E or NE, *station address*			
GT Information *Dest IP Net,* *Gateway IP, Hops*			
NRV Entries *node number,* *IP Address,* *LU, LU T/O,Level,* N, MA, *MA T/O*			

Worksheets and NSINIT Answer Files

This subsection contains Network Worksheets and NSINIT initialization answer files for the following configurations:

- a two-node LAN network, with NODE1 and NODE2
- a two-node Router/1000 Network, with NODE1 and NODE2.

Two-Node LAN

Figure A-1 illustrates the network map for the two-node LAN network used in this subsection. Both nodes support NS Common Services, ARPA Services, and DS/1000-IV Compatible Services.

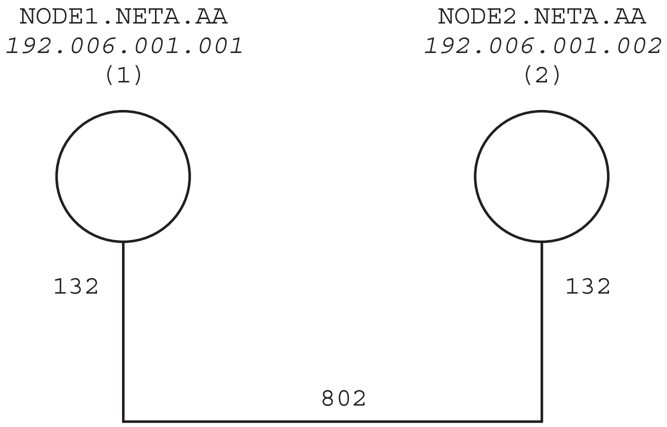


Figure A-1. Two-Node LAN

Example Network Worksheet (EX_LAN_WORKSHEET.TXT)

```
                       EX_LAN_WORKSHEET.TXT 91790-17055 REV.xxxx <930803.1053>
                                                2 node LAN with NS-ARPA and
                                                DS/1000-IV Compatible Services

                     NS-ARPA/1000  NETWORK  WORKSHEET
                                                   ---------------------------

Note: The same number of asterisks (*, **) indicates related information.
      () indicate default values.  /D indicates default value taken.
      For LAN/1000 links, user LU = even value = read LU - 1
      For all other links, transmit LU = receive LU - 1
      Node Number is equivalent to Router/1000 address.

      In this file, wherever LAN is mentioned, either 802 or ETHERNET
      is equally applicable.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
NODAL INFORMATION
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

1.Node Name            node1.neta.aa       node2.neta.aa
-------------------|-------------------|-------------------|------------------
2.IP Addresses       192.006.001.001     192.006.001.002
-------------------|-------------------|-------------------|------------------
3.Router/1000 Address        1                   2
-------------------|-------------------|-------------------|------------------
4.Files

NSINIT (NSOUTxx,
xx=1 to 99)          node1_lan.nsin      node2_lan.nsin
-------------------|-------------------|-------------------|------------------
NRINIT
(LU 1)               all_nodes.nrin      all_nodes.nrin
-------------------|-------------------|-------------------|------------------
NSTRC
(NS_TRACE.TRC)               /D                  /D
-------------------|-------------------|-------------------|------------------
FMTRC
(NS_TRACE.FMT)               /D                  /D
-------------------|-------------------|-------------------|------------------
EVMON
(/SYSTEM/NS_EVENT.LOG)       /D                  /D
-------------------|-------------------|-------------------|------------------
5. Link Information
Valid Link Interface | Communication link combinations:
          LAN        |      LAN
          RTR        |      HDLC, X.25 (RTE-RTE)
          none       |      X.25 (RTE-MPE) or Bisync
LAN Station addresses are in hexadecimal.
```

```
Communication Link          LAN                 LAN
-------------------|--------------------|--------------------|------------------
Link Interface type         LAN                 LAN
-------------------|--------------------|--------------------|------------------
  user or     read or
transmit LU, receive LU    96/97               96/97
-------------------|--------------------|--------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|--------------------|--------------------|------------------

Communication link
-------------------|--------------------|--------------------|------------------
Link Interface type
-------------------|--------------------|--------------------|------------------
  user or     read or
transmit LU, receive LU
-------------------|--------------------|--------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|--------------------|--------------------|------------------

Communication link
-------------------|--------------------|--------------------|------------------
Link Interface type
-------------------|--------------------|--------------------|------------------
  user or     read or
transmit LU, receive LU
-------------------|--------------------|--------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|--------------------|--------------------|------------------

** Pool LUs for
X.25 MPE links
-------------------|--------------------|--------------------|------------------
6.Network Membership

*** No. of networks to which
node is directly connected   1                   1
-------------------|--------------------|--------------------|------------------

7.Services

NFT (Y)                      Y                   Y
-------------------|--------------------|--------------------|------------------
*DS/1000-IV
Compatible Serv (N)          Y                   Y
-------------------|--------------------|--------------------|------------------
   1000 - 1000 (Y)           Y                   Y
   ----------------|--------------------|--------------------|------------------
 **1000 - 3000 (N)           N                   N
   ----------------|--------------------|--------------------|------------------
   All monitors (Y)
   VCPMN, TRFAS, RDBAM,
   RFAM, PROGL, PTOPM,
   OPERM, EXECW, EXECM,
   DLIST, CNSLM              Y                   Y
-------------------|-------------------|-------------------|------------------

8.Probe Information

Proxy Nodal Registry
Server (N)                   /D                  /D
-------------------|-------------------|-------------------|------------------
Target Address
(09-00-09-00-00-01)          /D                  /D
-------------------|-------------------|-------------------|------------------
Proxy  Address
(09-00-09-00-00-02)          /D                  /D
-------------------|-------------------|-------------------|------------------

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
ROUTING INFORMATION
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

9.***Directly Connected Network (DCN) Information (for RTR and LAN LIs)

IP Address           192.006.001.001     192.006.001.002
-------------------|-------------------|-------------------|------------------
RTR or LAN                  LAN                 LAN
-------------------|-------------------|-------------------|------------------
segsize (8000 if RTR,
1514 if LAN)        /D                  /D
-------------------|-------------------|-------------------|------------------
if LAN, user LU             96                  96
-------------------|-------------------|-------------------|------------------
if LAN, Enable/
Not Enable (E)              /D                  /D
-------------------|-------------------|-------------------|------------------

10. Gateway Table (GT) Information

Dest IP net
-------------------|-------------------|-------------------|------------------
Gateway IP
-------------------|-------------------|-------------------|------------------
Hops (100)
-------------------|-------------------|-------------------|------------------

11. * NRV Entries for remote nodes
    Make an entry for each remote node that:
      a)supports DS/1000-IV Compatible Services (RTE-RTE)
   or
      b)is connected via a RTR link interface.

   Where:
     Node Number is equivalent to Router/1000 address.
     Valid choices for Neighbor node are:  N or blank
     Valid choices for Message Accounting are:  MA or blank
```

```
Node Number                  2                   1
-------------------|-------------------|-------------------|------------------
IP address           192.006.001.002     192.006.001.001
-------------------|-------------------|-------------------|------------------
If no rerouting, LU         /D                  /D
-------------------|-------------------|-------------------|------------------
LU timeout (45 sec)         /D                  /D
-------------------|-------------------|-------------------|------------------
Level (1)                   /D                  /D
-------------------|-------------------|-------------------|------------------
Neighbor node               /D                  /D
-------------------|-------------------|-------------------|------------------
Message Accounting (no MA)   MA                  MA
-------------------|-------------------|-------------------|------------------
MA timeout (3)              /D                  /D
-------------------|-------------------|-------------------|------------------

Node Number
-------------------|-------------------|-------------------|------------------
IP address
-------------------|-------------------|-------------------|------------------
If no rerouting, LU
-------------------|-------------------|-------------------|------------------
LU timeout (45 sec)
-------------------|-------------------|-------------------|------------------
Level (1)
-------------------|-------------------|-------------------|------------------
Neighbor node
-------------------|-------------------|-------------------|------------------
Message Accounting (no MA)
-------------------|-------------------|-------------------|------------------
MA timeout (3)
-------------------|-------------------|-------------------|------------------

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
SECURITY CODES
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

12. Network Security Code   ns                  ns
-------------------|-------------------|-------------------|------------------
User’s Security Code        ns                  ns
-------------------|-------------------|-------------------|------------------
13.**Bisync Dial-Up
Local ID Sequence
-------------------|-------------------|-------------------|------------------
Remote ID Sequence
-------------------|-------------------|-------------------|------------------
```

Example NSINIT Answer File (`NODE1_LAN.NSIN`)

```
* NODE1_LAN.NSIN 91790-17062 REV.xxxx
*
* NSINIT file: NODE1_LAN.NSIN    5:06 PM  FRI.,  6  MAY , 1994. *
* Network Initialization Options:
*      1: Build Output File.
*      2: Build Output File & Initialize Network.
*      3: Initialize Network.
*      4: Shut Down Network Subsystem.
* Enter an option number [(1)..4]:
3   * Initialize Network *
* Enter the local node name.  Format: name.domain.organization
* (Each field may be 1..16 char).
* Local Name:
node1.neta.aa
*++   Classes of Events to log   ++*
*      6: Resource limit exceeded
*      5: Disaster (irrecoverable error)
*      4: Error    (severe, but recoverable error)
*      3: Warning  (unexpected event)
*      2: Event message
*      1: Internal State Information
*
* Enter the event classes to log, one per line.
* [/D = 4, 5, and 6].  Type /E to end.
* Event class:
/d
* Enter an event log file name [default = /system/ns_event.log]:
/d
* Do you want to start NSTRC when NS is enabled [Y/(N)]?
/d
* Do you want Network File Transfer (NFT) [(Y)/N]?
/d
*++  Network File Transfer (NFT) ++*
*   Default Buffer Size:      2048 Bytes
*   Transport Checksum Used:  No
*
* Do you want to modify these values [Y/(N)]?
/d
*
*++   DS/1000-IV Compatible Services    ++*
*
* Do you want any DS/1000-IV compatible services [Y/(N)]?
y
* Enter the maximum number of concurrent DS/1000-IV transactions on this
* node [1..(20)..100]:
/d
* Enter the maximum number of concurrent DS/1000-IV remote transactions to
* access this node [1..(7)..253]:
/d
*++   DS/1000-IV Timeout Values    ++*
* Master timeout in seconds:      45
* Slave timeout in seconds:       30
* Idle ”session” timer in hours:   5
*
* Do you want to modify these values [Y/(N)]?
/d
* Do you want any DS/1000–IV 1000–1000 compatible services [(Y)/N] ?
/d
* Do you want DS/1000–IV 1000–3000 services [Y/(N)]?
/d
* Should NSINIT try to schedule all the DS/1000–IV monitors (VCPMN, TRFAS
* RDBAM, RFAM, PROGL, PTOPM, OPERM, EXECW, EXECM, DLIST) [(Y)/N]?
/d
*++   NS Nodal Information   ++*
*
* Defaults are derived from previous responses.
*     Maximum number of active NS programs is: 24
*     Maximum number of active NS sockets is:  77
*
* Do you want to modify these values [Y/(N)]?
/d
*++          Nodal Registry          ++*
*
*     Maximum number of Connect–Site path reports is:         101
*      (The above value is derived from previous responses.)
*     Maximum number of Nodal path reports is:                20
*
* Do you want to modify these values [Y/(N)]?
/d
* Enter the maximum number of name records (the default is derived
* from previous responses) [1..(50)..303]:
/d
*++    Transmission Control Protocol (TCP)  ++*
*
* Initial Segment Size in bytes:                  4096
* Retransmission Backoff Algorithm:               Exponential
* Retransmission Smoothing parameter Alpha:       9
* Retransmission Smoothing parameter Beta:        20
* Do you want to modify these values? [Y/(N)]:
/d
*++ DCN ++*
* Enter information on this node for each Directly Connected Network.
* Format:
*
* <local IP addr>, <[subnet mask],> RTR, <segsize: [1200..(8000)]>
* <local IP addr>, <[subnet mask],> 802/LAN/ETHERNET,
* <segsize:[1200..(1514)]>, <link LU>, (E)/NE, <[station addr]>
*
* Where E = Enable, NE = do Not Enable.  Type /E to end.
*
*DCN:
192.006.001.001,,LAN,,96
*DCN:
/e
** Internal Use Only **
/e
```

```
*++ GT ++*
* Enter information for each Gateway (GT).
* Format:
*
* <Dest IP Net>, <Gateway IP>, <Hops: [1..(100)]>
*
* Where: Dest IP Net = IP Address of any node on a remote net
*        Gateway IP  = DCN IP Address of Gateway to use
*        Hops        = Maximum allowed IP hops to Dest IP Net
* Type /E to end.
*
*GT:
/e
* Enter the maximum number of Path Records for IP. The default is derived
* from previous responses [6..(103)..200]:
/d
*+++   ADDRESS RESOLUTION   +++*
* Maximum number of active PCB Records:     10
* Retry interval timeout in centi-seconds:  100
* Proxy Nodal Registry Server:              No
*
* Do you want to modify these values [Y/(N)]:
/d
*
* Default IEEE-802 Multicast Addresses for Probe:
*    Target Address: 09-00-09-00-00-01
*    Proxy Address:  09-00-09-00-00-02
*
* Do you want to modify these values [Y/(N)]?
/d
*
*++      Router/1000      ++*
*
* Enter the Router/1000 address of the local node.[1..32767]:
1
* Enter the MA retry limit [1..(4)..13]:
/d
* Enter the Nodal Routing Vector (NRV) information. Type /E to end. Format:
* <Node Number>, <IP addr>, <LU>, <LU T/O>, <Level>, N, MA, <MA T/O>
*
* Where:
*    node number: Router/1000 node address
*    IP Address : IP Address. Format: n.n.n.n where n = 1..255
*    LU         : Link LU if rerouting is not used to access the node
*    LU T/O     : Link timeout in seconds
*    level      : DS/1000 software level
*    N          : Neighbor bit, set if this node is a neighbor to local node
*    MA         : Use Message Accounting in transactions to this node
*    MA T/O     : Message Accounting timeout in seconds
*
*NRV:
2,192.006.001.002,,,,,ma
*NRV:
/e
```

```
* Enter the LU information for Router/1000. Type /E to end.
* Format: <Link LU>, (E)/NE, (RR)/NR, <cost: [(1)..99]>
*
* Where:
*    E = Enable, NE = do Not Enable
*   RR = use ReRouting, NR = do Not use ReRouting
* Link:
/e
* Enter the Network security code for this node [1..2 char]:
ns
* Enter the Network User’s security code for this node [1..2 char]:
ns
*++   EMA Usage   ++*
*  3 pages available   1 pages used
*+++ System resources required for this file +++*
* DSAM Table size in words:               32166
* Number of RTE Class Numbers:               19
* Number of RTE Resource Numbers:             6
* SMB size in words:                        170
* NSINIT successfully completed action BUILD.
```

Example NSINIT Answer File (NODE2_LAN.NSIN)

```
* NODE2_LAN.NSIN 91790-17051 REV.xxxx
*
* NSINIT file: NODE2_LAN.NSIN    5:06 PM  FRI.,  6  MAY , 1994. *
* Network Initialization Options:
*      1: Build Output File.
*      2: Build Output File & Initialize Network.
*      3: Initialize Network.
*      4: Shut Down Network Subsystem.
* Enter an option number [(1)..4]:
3   * Initialize Network *
* Enter the local node name.  Format: name.domain.organization
* (Each field may be 1..16 char).
* Local Name:
node2.neta.aa
*++   Classes of Events to log   ++*
*      6: Resource limit exceeded
*      5: Disaster (irrecoverable error)
*      4: Error    (severe, but recoverable error)
*      3: Warning  (unexpected event)
*      2: Event message
*      1: Internal State Information
*
* Enter the event classes to log, one per line.
* [/D = 4, 5, and 6].  Type /E to end.
* Event class:
/d
* Enter an event log file name [default = /system/ns_event.log]:
/d
* Do you want to start NSTRC when NS is enabled [Y/(N)]?
/d
* Do you want Network File Transfer (NFT) [(Y)/N]?
/d
*++  Network File Transfer (NFT) ++*
*   Default Buffer Size:      2048 Bytes
*   Transport Checksum Used:  No
*
* Do you want to modify these values [Y/(N)]?
/d
*
*++   DS/1000-IV Compatible Services    ++*
*
* Do you want any DS/1000-IV compatible services [Y/(N)]?
y
* Enter the maximum number of concurrent DS/1000-IV transactions on this
* node [1..(20)..100]:
/d
* Enter the maximum number of concurrent DS/1000-IV remote transactions to
* access this node [1..(7)..253]:
/d
*++   DS/1000-IV Timeout Values    ++*
* Master timeout in seconds:      45
* Slave timeout in seconds:       30
* Idle ”session” timer in hours:   5
*
* Do you want to modify these values [Y/(N)]?
/d
* Do you want any DS/1000-IV 1000-1000 compatible services [(Y)/N] ?
/d
* Do you want DS/1000-IV 1000-3000 services [Y/(N)]?
/d
* Should NSINIT try to schedule all the DS/1000-IV monitors (VCPMN, TRFAS
* RDBAM, RFAM, PROGL, PTOPM, OPERM, EXECW, EXECM, DLIST) [(Y)/N]?
/d
*++   NS Nodal Information   ++*
*
* Defaults are derived from previous responses.
*     Maximum number of active NS programs is: 24
*     Maximum number of active NS sockets is:  77
*
* Do you want to modify these values [Y/(N)]?
/d
*++          Nodal Registry          ++*
*
*     Maximum number of Connect-Site path reports is:         101
*      (The above value is derived from previous responses.)
*     Maximum number of Nodal path reports is:                20
*
* Do you want to modify these values [Y/(N)]?
/d
* Enter the maximum number of name records (the default is derived
* from previous responses) [1..(50)..303]:
/d
*++    Transmission Control Protocol (TCP)  ++*
*
* Initial Segment Size in bytes:                 4096
* Retransmission Backoff Algorithm:              Exponential
* Retransmission Smoothing parameter Alpha:      9
* Retransmission Smoothing parameter Beta:       20
* Do you want to modify these values? [Y/(N)]:
/d
*++ DCN ++*
* Enter information on this node for each Directly Connected Network.
* Format:
*
* <local IP addr>, <[subnet mask],> RTR, <segsize: [1200..(8000)]>
* <local IP addr>, <[subnet mask],> 802/LAN/ETHERNET,
* <segsize:[1200..(1514)]>, <link LU>, (E)/NE, <[station addr]>
*
* Where E = Enable, NE = do Not Enable.  Type /E to end.
*
*DCN:
192.006.001.002,,LAN,,96
*DCN:
/e
** Internal Use Only **
/e
```

```
*++ GT ++*
* Enter information for each Gateway (GT).
* Format:
*
* <Dest IP Net>, <Gateway IP>, <Hops: [1..(100)]>
*
* Where: Dest IP Net = IP Address of any node on a remote net
*        Gateway IP  = DCN IP Address of Gateway to use
*        Hops        = Maximum allowed IP hops to Dest IP Net
* Type /E to end.
*
*GT:
/e
* Enter the maximum number of Path Records for IP. The default is derived
* from previous responses [6..(103)..200]:
/d
*+++   ADDRESS RESOLUTION   +++*
* Maximum number of active PCB Records:     10
* Retry interval timeout in centi-seconds:  100
* Proxy Nodal Registry Server:              No
*
* Do you want to modify these values [Y/(N)]:
/d
*
* Default IEEE-802 Multicast Addresses for Probe:
*    Target Address: 09-00-09-00-00-01
*    Proxy Address:  09-00-09-00-00-02
*
* Do you want to modify these values [Y/(N)]?
/d
*
*++      Router/1000      ++*
*
* Enter the Router/1000 address of the local node.[1..32767]:
2
* Enter the MA retry limit [1..(4)..13]:
/d
* Enter the Nodal Routing Vector (NRV) information. Type /E to end. Format:
* <Node Number>, <IP addr>, <LU>, <LU T/O>, <Level>, N, MA, <MA T/O>
*
* Where:
*    node number: Router/1000 node address
*    IP Address : IP Address. Format: n.n.n.n where n = 1..255
*    LU         : Link LU if rerouting is not used to access the node
*    LU T/O     : Link timeout in seconds
*    level      : DS/1000 software level
*    N          : Neighbor bit, set if this node is a neighbor to local node
*    MA         : Use Message Accounting in transactions to this node
*    MA T/O     : Message Accounting timeout in seconds
*
*NRV:
1,192.006.001.001,,,,,ma
*NRV:
/e
```

```
* Enter the LU information for Router/1000. Type /E to end.
* Format: <Link LU>, (E)/NE, (RR)/NR, <cost: [(1)..99]>
*
* Where:
*    E = Enable, NE = do Not Enable
*   RR = use ReRouting, NR = do Not use ReRouting
* Link:
/e
* Enter the Network security code for this node [1..2 char]:
ns
* Enter the Network User’s security code for this node [1..2 char]:
ns
*++   EMA Usage   ++*
*  3 pages available   1 pages used
*+++ System resources required for this file +++*
* DSAM Table size in words:               32166
* Number of RTE Class Numbers:               19
* Number of RTE Resource Numbers:             6
* SMB size in words:                        170
* NSINIT successfully completed action BUILD.
```

Two-Node Router/1000 Network

Figure A-2 illustrates the network map for the two-node Router/1000 network used in this subsection. Both nodes support NS Common Services, ARPA Services, and DS/1000-IV Compatible Services.

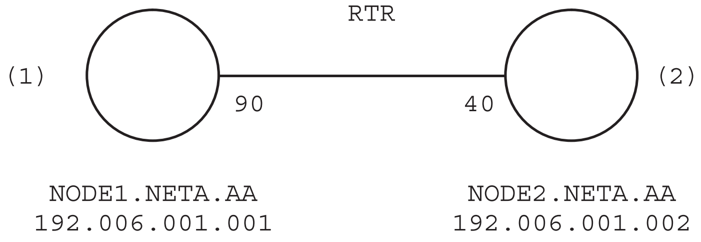


Figure A-2. Two-Node Router/1000 Network

Example Network Worksheet (EX_RTR_WORKSHEET.TXT)

```
                       EX_RTR_WORKSHEET.TXT 91790-17056 REV.xxxx <930803.1057>
                                                2 node RTR with NS-ARPA and
                                                DS/1000-IV Compatible Services

                        NS-ARPA/1000  NETWORK  WORKSHEET
                        ---------------------------

Note: The same number of asterisks (*, **) indicates related information.
      () indicate default values.  /D indicates default value taken.
      For LAN/1000 links, user LU = even value = read LU - 1
      For all other links, transmit LU = receive LU - 1.
      Node Number is equivalent to Router/1000 address.

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
NODAL INFORMATION
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

1.Node Name            node1.neta.aa       node2.neta.aa
-------------------|-------------------|-------------------|------------------
2.IP Addresses       192.006.001.001     192.006.001.002
-------------------|-------------------|-------------------|------------------
3.Router/1000 Address        1                   2
-------------------|-------------------|-------------------|------------------
4.Files

NSINIT (NSOUTxx,
xx=1 to 99)          node1_rtr.nsin      node2_rtr.nsin
-------------------|-------------------|-------------------|------------------
NRINIT
(LU 1)               all_nodes.nrin      all_nodes.nrin
-------------------|-------------------|-------------------|------------------
NSTRC
(NS_TRACE.TRC)               /D                  /D
-------------------|-------------------|-------------------|------------------
FMTRC
(NS_TRACE.FMT)               /D                  /D
-------------------|-------------------|-------------------|------------------
EVMON
(/SYSTEM/NS_EVENT.LOG)       /D                  /D
-------------------|-------------------|-------------------|------------------
5. Link Information
   Valid Link Interface | Communication link combinations:
             LAN        |      LAN
             RTR        |      HDLC, X.25 (RTE-RTE)
             none       |      X.25 (RTE-MPE) or Bisync
LAN Station addresses are in hexadecimal.

Communication Link          HDLC                HDLC
-------------------|-------------------|-------------------|------------------
Link Interface type         rtr                 rtr
-------------------|-------------------|-------------------|------------------
  user or     read or
transmit LU, receive LU     79/80               79/80
```

```
-------------------|-------------------|-------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|-------------------|-------------------|------------------

Communication link
-------------------|-------------------|-------------------|------------------
Link Interface type
-------------------|-------------------|-------------------|------------------
  user or     read or
transmit LU, receive LU
-------------------|-------------------|-------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|-------------------|-------------------|------------------

Communication link
-------------------|-------------------|-------------------|------------------
Link Interface type
-------------------|-------------------|-------------------|------------------
  user or     read or
transmit LU, receive LU
-------------------|-------------------|-------------------|------------------
LAN Station Address
08-00-09-hh-hh-hh
-------------------|-------------------|-------------------|------------------

** Pool LUs for
X.25 MPE links
-------------------|-------------------|-------------------|------------------
6.Network Membership

*** No. of networks to which
node is directly connected   1                   1
-------------------|-------------------|-------------------|------------------

7.Services

NFT (Y)                      Y                   Y
-------------------|-------------------|-------------------|------------------
*DS/1000-IV
Compatible Serv (N)          Y                   Y
-------------------|-------------------|-------------------|------------------
   1000 - 1000 (Y)           Y                   Y
   ----------------|-------------------|-------------------|------------------
 **1000 - 3000 (N)           N                   N
   ----------------|-------------------|-------------------|------------------
  All monitors (Y)
  VCPMN, TRFAS, RDBAM,
  RFAM, PROGL, PTOPM,
  OPERM, EXECW, EXECM,
  DLIST, CNSLM              Y                   Y
-------------------|-------------------|-------------------|------------------
```

```
8.Probe Information

Proxy Nodal Registry
Server (N)
-------------------|-------------------|-------------------|------------------
Target Address
(09-00-09-00-00-01)
-------------------|-------------------|-------------------|------------------
Proxy  Address
(09-00-09-00-00-02)
-------------------|-------------------|-------------------|------------------

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
ROUTING INFORMATION
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

9.***Directly Connected Network (DCN) Information (for RTR and LAN LIs)

IP Address           192.006.001.001     192.006.001.002
-------------------|-------------------|-------------------|------------------
RTR or LAN                  rtr                 rtr
-------------------|-------------------|-------------------|------------------
segsize (8000 if RTR,
         1500 if LAN)        /D                  /D
-------------------|-------------------|-------------------|------------------
if LAN, user LU
-------------------|-------------------|-------------------|------------------
if LAN, Enable/
Not Enable (E)
-------------------|-------------------|-------------------|------------------

10. Gateway Table (GT) Information

Dest IP net
-------------------|-------------------|-------------------|------------------
Gateway IP
-------------------|-------------------|-------------------|------------------
Hops (100)
-------------------|-------------------|-------------------|------------------

11. * NRV Entries for remote nodes
    Make an entry for each remote node that:
      a)supports DS/1000-IV Compatible Services (RTE-RTE)
    or
      b)is connected via a RTR link interface.

    Where:
      Node Number is equivalent to Router/1000 address.
      Valid choices for Neighbor node are:  N or blank
      Valid choices for Message Accounting are:  MA or blank

Node Number                  2                   1
-------------------|-------------------|-------------------|------------------
IP address           192.006.001.002     192.006.001.001
-------------------|-------------------|-------------------|------------------
```

```
If no rerouting, LU         /D                  /D
-------------------|-------------------|-------------------|------------------
LU timeout (45 sec)         /D                  /D
-------------------|-------------------|-------------------|------------------
Level (1)                   /D                  /D
-------------------|-------------------|-------------------|------------------
Neighbor node               /D                  /D
-------------------|-------------------|-------------------|------------------
Message Accounting (no MA)   MA                  MA
-------------------|-------------------|-------------------|------------------
MA timeout (3)              /D                  /D
-------------------|-------------------|-------------------|------------------

Node Number
-------------------|-------------------|-------------------|------------------
IP address
-------------------|-------------------|-------------------|------------------
If no rerouting, LU
-------------------|-------------------|-------------------|------------------
LU timeout (45 sec)
-------------------|-------------------|-------------------|------------------
Level (1)
-------------------|-------------------|-------------------|------------------
Neighbor node
-------------------|-------------------|-------------------|------------------
Message Accounting (no MA)
-------------------|-------------------|-------------------|------------------
MA timeout (3)
-------------------|-------------------|-------------------|------------------

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
SECURITY CODES
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

12. Network Security Code   ns                  ns
-------------------|-------------------|-------------------|------------------
User’s Security Code        ns                  ns
-------------------|-------------------|-------------------|------------------
13.**Bisync Dial-Up
Local ID Sequence
-------------------|-------------------|-------------------|------------------
Remote ID Sequence
-------------------|-------------------|-------------------|------------------
```

Example NSINIT Answer File (`NODE1_RTR.NSIN`)

```
* NODE1_RTR.NSIN 91790-17050 REV.xxxx
*
* NSINIT file: NODE1_RTR.NSIN    5:06 PM  FRI.,  6  MAY , 1994. *
* Network Initialization Options:
*      1: Build Output File.
*      2: Build Output File & Initialize Network.
*      3: Initialize Network.
*      4: Shut Down Network Subsystem.
* Enter an option number [(1)..4]:
3   * Initialize Network *
* Enter the local node name.  Format: name.domain.organization
* (Each field may be 1..16 char).
* Local Name:
node1.neta.aa
*++   Classes of Events to log   ++*
*      6: Resource limit exceeded
*      5: Disaster (irrecoverable error)
*      4: Error    (severe, but recoverable error)
*      3: Warning  (unexpected event)
*      2: Event message
*      1: Internal State Information
*
* Enter the event classes to log, one per line.
* [/D = 4, 5, and 6].  Type /E to end.
* Event class:
/d
* Enter an event log file name [default = /system/ns_event.log]:
/d
* Do you want to start NSTRC when NS is enabled [Y/(N)]?
/d
* Do you want Network File Transfer (NFT) [(Y)/N]?
/d
*++  Network File Transfer (NFT) ++*
*   Default Buffer Size:      2048 Bytes
*   Transport Checksum Used:  No
*
* Do you want to modify these values [Y/(N)]?
/d
*
*++   DS/1000-IV Compatible Services    ++*
*
* Do you want any DS/1000-IV compatible services [Y/(N)]?
y
* Enter the maximum number of concurrent DS/1000-IV transactions on this
* node [1..(20)..100]:
/d
* Enter the maximum number of concurrent DS/1000-IV remote transactions to
* access this node [1..(7)..253]:
/d
*++   DS/1000-IV Timeout Values    ++*
* Master timeout in seconds:      45
* Slave timeout in seconds:       30
* Idle ”session” timer in hours:   5
*
* Do you want to modify these values [Y/(N)]?
/d
* Do you want any DS/1000-IV 1000-1000 compatible services [(Y)/N] ?
/d
* Do you want DS/1000-IV 1000-3000 services [Y/(N)]?
/d
* Should NSINIT try to schedule all the DS/1000-IV monitors (VCPMN, TRFAS
* RDBAM, RFAM, PROGL, PTOPM, OPERM, EXECW, EXECM, DLIST) [(Y)/N]?
/d
*++   NS Nodal Information   ++*
*
* Defaults are derived from previous responses.
*     Maximum number of active NS programs is: 24
*     Maximum number of active NS sockets is:  77
*
* Do you want to modify these values [Y/(N)]?
/d
*++          Nodal Registry          ++*
*
*     Maximum number of Connect-Site path reports is:         101
*      (The above value is derived from previous responses.)
*     Maximum number of Nodal path reports is:                20
*
* Do you want to modify these values [Y/(N)]?
/d
* Enter the maximum number of name records (the default is derived
* from previous responses) [1..(50)..303]:
/d
*++    Transmission Control Protocol (TCP)  ++*
*
* Initial Segment Size in bytes:                  4096
* Retransmission Backoff Algorithm:               Exponential
* Retransmission Smoothing parameter Alpha:       9
* Retransmission Smoothing parameter Beta:        20
* Do you want to modify these values? [Y/(N)]:
/d
*++ DCN ++*
* Enter information on this node for each Directly Connected Network.
* Format:
*
* <local IP addr>, <[subnet mask],> RTR, <segsize: [1200..(8000)]>
* <local IP addr>, <[subnet mask],> 802/LAN/ETHERNET,
* <segsize:[1200..(1514)]>, <link LU>, (E)/NE, <[station addr]>
*
* Where E = Enable, NE = do Not Enable.  Type /E to end.
*
*DCN:
192.006.001.001,,RTR
*DCN:
/e
** Internal Use Only **
/e
```

```
*++ GT ++*
* Enter information for each Gateway (GT).
* Format:
*
* <Dest IP Net>, <Gateway IP>, <Hops: [1..(100)]>
*
* Where: Dest IP Net = IP Address of any node on a remote net
*        Gateway IP  = DCN IP Address of Gateway to use
*        Hops        = Maximum allowed IP hops to Dest IP Net
* Type /E to end.
*
*GT:
/e
* Enter the maximum number of Path Records for IP. The default is derived
* from previous responses [6..(103)..200]:
/d
*
*++      Router/1000      ++*
*
* Enter the Router/1000 address of the local node.[1..32767]:
1
* Enter the MA retry limit [1..(4)..13]:
/d
* Enter the Nodal Routing Vector (NRV) information. Type /E to end. Format:
* <Node Number>, <IP addr>, <LU>, <LU T/O>, <Level>, N, MA, <MA T/O>
*
* Where:
*    node number: Router/1000 node address
*    IP Address : IP Address. Format: n.n.n.n where n = 1..255
*    LU         : Link LU if rerouting is not used to access the node
*    LU T/O     : Link timeout in seconds
*    level      : DS/1000 software level
*    N          : Neighbor bit, set if this node is a neighbor to local node
*    MA         : Use Message Accounting in transactions to this node
*    MA T/O     : Message Accounting timeout in seconds
*
*NRV:
2,192.006.001.002,,,,,ma
*NRV:
/e
* Enter the LU information for Router/1000. Type /E to end.
* Format: <Link LU>, (E)/NE, (RR)/NR, <cost: [(1)..99]>
*
* Where:
*    E = Enable, NE = do Not Enable
*   RR = use ReRouting, NR = do Not use ReRouting
* Link:
79
* Link:
/e
* Enter the Network security code for this node [1..2 char]:
ns
* Enter the Network User's security code for this node [1..2 char]:
ns
```

```
*++   EMA Usage   ++*
*  3 pages available   1 pages used
*+++ System resources required for this file +++*
* DSAM Table size in words:               30527
* Number of RTE Class Numbers:               19
* Number of RTE Resource Numbers:             6
* SMB size in words:                        170
* NSINIT successfully completed action BUILD.
```

Example NSINIT Answer File (NODE2_RTR.NSIN)

```
* NODE2_RTR.NSIN 91790-17068 REV.xxxx
*
* NSINIT file: NODE2_RTR.NSIN    5:06 PM  FRI.,  6  MAY , 1994. *
* Network Initialization Options:
*      1: Build Output File.
*      2: Build Output File & Initialize Network.
*      3: Initialize Network.
*      4: Shut Down Network Subsystem.
* Enter an option number [(1)..4]:
3   * Initialize Network *
* Enter the local node name.  Format: name.domain.organization
* (Each field may be 1..16 char).
* Local Name:
node2.neta.aa
*++   Classes of Events to log   ++*
*      6: Resource limit exceeded
*      5: Disaster (irrecoverable error)
*      4: Error    (severe, but recoverable error)
*      3: Warning  (unexpected event)
*      2: Event message
*      1: Internal State Information
*
* Enter the event classes to log, one per line.
* [/D = 4, 5, and 6].  Type /E to end.
* Event class:
/d
* Enter an event log file name [default = /system/ns_event.log]:
/d
* Do you want to start NSTRC when NS is enabled [Y/(N)]?
/d
* Do you want Network File Transfer (NFT) [(Y)/N]?
/d
*++  Network File Transfer (NFT) ++*
*   Default Buffer Size:      2048 Bytes
*   Transport Checksum Used:  No
*
* Do you want to modify these values [Y/(N)]?
/d
*
*++   DS/1000-IV Compatible Services    ++*
*
* Do you want any DS/1000-IV compatible services [Y/(N)]?
y
* Enter the maximum number of concurrent DS/1000-IV transactions on this
* node [1..(20)..100]:
/d
* Enter the maximum number of concurrent DS/1000-IV remote transactions to
* access this node [1..(7)..253]:
/d
*++   DS/1000-IV Timeout Values    ++*
* Master timeout in seconds:      45
* Slave timeout in seconds:       30
* Idle ”session” timer in hours:   5
*
* Do you want to modify these values [Y/(N)]?
/d
* Do you want any DS/1000-IV 1000-1000 compatible services [(Y)/N] ?
/d
* Do you want DS/1000-IV 1000-3000 services [Y/(N)]?
/d
* Should NSINIT try to schedule all the DS/1000-IV monitors (VCPMN, TRFAS
* RDBAM, RFAM, PROGL, PTOPM, OPERM, EXECW, EXECM, DLIST) [(Y)/N]?
/d
*++   NS Nodal Information   ++*
*
* Defaults are derived from previous responses.
*     Maximum number of active NS programs is: 24
*     Maximum number of active NS sockets is:  77
*
* Do you want to modify these values [Y/(N)]?
/d
*++          Nodal Registry          ++*
*
*     Maximum number of Connect-Site path reports is:         101
*      (The above value is derived from previous responses.)
*     Maximum number of Nodal path reports is:                20
*
* Do you want to modify these values [Y/(N)]?
/d
* Enter the maximum number of name records (the default is derived
* from previous responses) [1..(50)..303]:
/d
*++    Transmission Control Protocol (TCP)  ++*
*
* Initial Segment Size in bytes:                  4096
* Retransmission Backoff Algorithm:               Exponential
* Retransmission Smoothing parameter Alpha:       9
* Retransmission Smoothing parameter Beta:        20
* Do you want to modify these values? [Y/(N)]:
/d
*++ DCN ++*
* Enter information on this node for each Directly Connected Network.
* Format:
*
* <local IP addr>, <[subnet mask],> RTR, <segsize: [1200..(8000)]>
* <local IP addr>, <[subnet mask],> 802/LAN/ETHERNET,
* <segsize:[1200..(1514)]>, <link LU>, (E)/NE, <[station addr]>
*
* Where E = Enable, NE = do Not Enable.  Type /E to end.
*
*DCN:
192.006.001.002,,rtr
*DCN:
/e
** Internal Use Only **
/e
```

```
*++ GT ++*
* Enter information for each Gateway (GT).
* Format:
*
* <Dest IP Net>, <Gateway IP>, <Hops: [1..(100)]>
*
* Where: Dest IP Net = IP Address of any node on a remote net
*        Gateway IP  = DCN IP Address of Gateway to use
*        Hops        = Maximum allowed IP hops to Dest IP Net
* Type /E to end.
*
*GT:
/e
* Enter the maximum number of Path Records for IP. The default is derived
* from previous responses [6..(103)..200]:
/d
*
*++      Router/1000      ++*
*
* Enter the Router/1000 address of the local node.[1..32767]:
2
* Enter the MA retry limit [1..(4)..13]:
/d
* Enter the Nodal Routing Vector (NRV) information. Type /E to end. Format:
* <Node Number>, <IP addr>, <LU>, <LU T/O>, <Level>, N, MA, <MA T/O>
*
* Where:
*    node number: Router/1000 node address
*    IP Address : IP Address. Format: n.n.n.n where n = 1..255
*    LU         : Link LU if rerouting is not used to access the node
*    LU T/O     : Link timeout in seconds
*    level      : DS/1000 software level
*    N          : Neighbor bit, set if this node is a neighbor to local node
*    MA         : Use Message Accounting in transactions to this node
*    MA T/O     : Message Accounting timeout in seconds
*
*NRV:
1,192.006.001.001,,,,,ma
*NRV:
/e
* Enter the LU information for Router/1000. Type /E to end.
* Format: <Link LU>, (E)/NE, (RR)/NR, <cost: [(1)..99]>
*
* Where:
*    E = Enable, NE = do Not Enable
*   RR = use ReRouting, NR = do Not use ReRouting
* Link:
79
* Link:
/e
* Enter the Network security code for this node [1..2 char]:
ns
* Enter the Network User’s security code for this node [1..2 char]:
ns
```

```
*++   EMA Usage   ++*
*  3 pages available   1 pages used
*+++ System resources required for this file +++*
* DSAM Table size in words:               30527
* Number of RTE Class Numbers:               19
* Number of RTE Resource Numbers:             6
* SMB size in words:                        170
* NSINIT successfully completed action BUILD.
```

NRINIT Input File (ALL_NODES.NRIN)

```
*
* ALL_NODES.NRIN 91790-17061 REV.xxxx <910709.1031>
* SOURCE: 91790-17061
*
* Sample NRINIT file for 2 sample NSINIT examples: 2 node LAN
*                                                  2 node Router/1000
*
* NPRs for the first node is the same for all three networks.
* Note that the LAN station address is defaulted, to make sure
* the same NPR can be used for node 1.
*
* Note that entries are included for ALL nodes so that this file can be
* used at any node without modification.  NRINIT will issue a warning when it
* encounters an entry for a local node and ignore that entry.
*
BEGIN node1.neta.aa 192.006.001.001 ip END
*
* Second NPR is the same for LAN and for Router/1000.
*
BEGIN node2.neta.aa 192.006.001.002 ip END
```

B

Point-to-Point Network Topologies

This appendix discusses topology considerations for point-to-point networks.

Redundant Links

Redundant links (Figure B-1) are useful for Router/1000 networks that are configured with Dynamic Rerouting links.

Redundant links provide alternate routes between nodes. If a node or link fails, the nodes can still communicate by rerouting around the failure. In a Router/1000 network with Dynamic Rerouting, the rerouting is automatic; otherwise you must explicitly configure new routes around the failure.

String

The string topology requires one less communication link than there are computers in the network, and so requires the fewest number of links. For communication between non-adjacent nodes, messages are stored and forwarded by intervening nodes. If a link fails, the nodes separated by the failure will not be able to communicate. For example, in Figure B-2, if the link in the string network between Node 1 and Node 2 fails, Node 1 will be unable to communicate with Nodes 2, 3, and 4.

Ring

The ring topology is a string topology with an additional link between the end nodes. The store-and-forward delay is half that of a string topology because the maximum number of intervening nodes is halved. The ring topology is suited for data-sharing applications in which databases stored at various nodes are accessible from all nodes.

Ring networks are less vulnerable than string networks. If any one link fails, all the nodes can still communicate by rerouting around the failure. If the ring is used in a Router/1000 network with Dynamic Rerouting, the rerouting is automatic; otherwise you must explicitly configure new routes around the failure. If a link fails, the store-and-forward delay will then be the same as a string.

Star

The star topology is often used for centralized data collection, supervisory control, or in an application where the outlying nodes have little storage capacity. It is also used when a central node has a large database or control program that is accessed by the other nodes. In a star network, there is at most one intervening node between any two nodes.

Star networks are vulnerable to failure of the central node. If the central node fails, no network communication is possible.

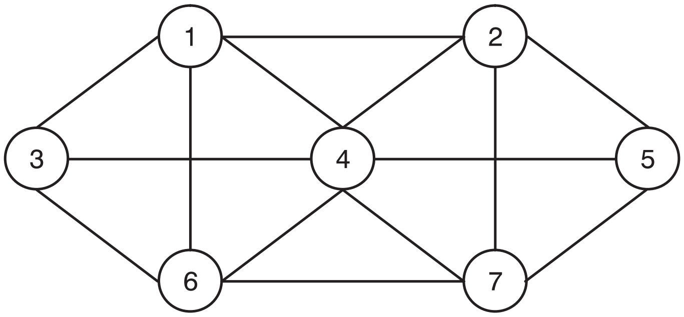
Hierarchical

The hierarchical topology is sometimes used with supervisory-control applications, where large databases exist at one node, possibly along with control programs, that are accessed by nodes lower in the hierarchy.

Commands and control setpoints can be computed at the highest level for specific interpretation by each of the lower nodes. Or, the network may divide the intelligence or control among the nodes so that a failure of any of the higher-level nodes will allow continued operation by the lower-level nodes.

You can also use hierarchical topologies for distributed database applications. For example, in Figure B-3, data can be stored at Node 2 for use by Nodes 4 and 5.

A failure in the lower-level nodes only affects the area it controls. This topology is the least vulnerable to failures of any except those with redundant links.



Redundant

Figure B-1. Redundant Links

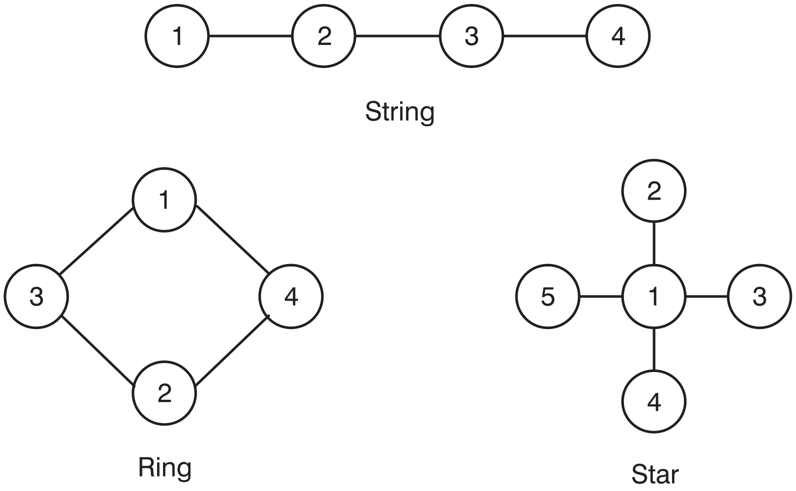


Figure B-2. String, Ring, and Star Topologies

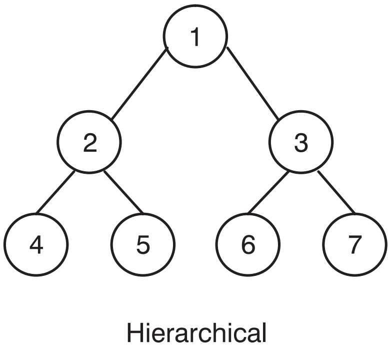


Figure B-3. Hierarchical Topology

Dynamic Rerouting Nodes and Non-Rerouting Nodes

Dynamic Rerouting nodes are compatible with non-rerouting nodes; that is, traffic between them may exist. However, Dynamic Rerouting will only occur between nodes that have Dynamic Rerouting software. In Figure B-4, R indicates Dynamic Rerouting nodes and N indicates non-rerouting nodes:

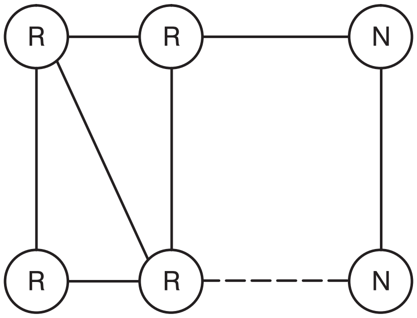


Figure B-4. Rerouting Topology Considerations

Dynamic rerouting can only occur within the R nodes. The extra link to N (denoted by the broken line) does not provide any more reliability for the network because the software can only use a fixed path as long as the node on the other side of a link is a non-rerouting node. A manual switch of the LU using the DSMOD CN command can restore communications on the non-rerouting nodes.

For a link to be a rerouting link, both nodes on either side must be Dynamic Rerouting nodes.

However, while compatible in the network, there are some topology restrictions that make it inadvisable to leave an un-upgraded DS/1000 node, or one without Dynamic Rerouting, in critical locations in the network. There are cases where a non-rerouting node can cause some problems with transmission being possible in one direction but not the other after a topology change. In Figure B-5, N indicates non-rerouting or DS/1000 nodes and R indicates a Dynamic Rerouting node.

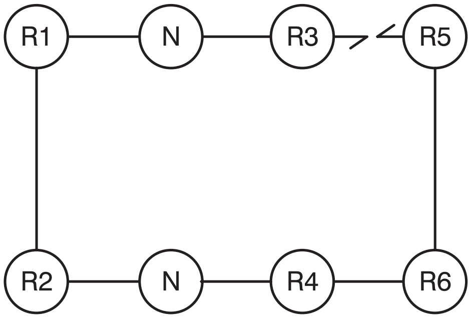


Figure B-5. Rerouting Topology Considerations

If the link between Dynamic Rerouting nodes R3 and R5 fails, requests will be able to go from R5 to R1 but replies will not flow from R1 to R5. This is because routing updates generated by R5 indicating link R3 to R5 is down are rejected as bad requests by N nodes, thereby preventing R1 and R2 from updating their tables. Since R2 and R1 can not receive routing messages from the rest of the network, they effectively are non-rerouting nodes also. When the failure of R3 to R5 occurs, R5 reroutes through R6 to R4 to N and gets its messages to R1. R1, having not received routing information, continues to use the R1 to N to R3 path to get to R5.

Since higher-level software (for example, MA) will eventually prevent requests from going to R1, the Network Manager is cautioned to set up a mixed network carefully.

HP recommends that you set up your network such that non-rerouting nodes are end nodes. The example in Figure B-6 below would work well and shows many sub-networks which would also work.

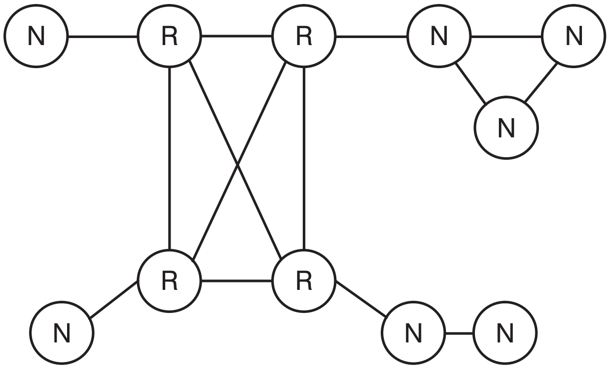


Figure B-6. Rerouting Topology Considerations

C

Planning A Memory-Based NS-ARPA System

Overview

This appendix explains what you need to consider when planning a memory-based NS-ARPA system.

A memory-based system is a system executing entirely in memory. It is a system that does not require access to a local or remote disk for its operation. The memory-based system is downloaded to a remote system for execution. This is done from the local system either interactively or at the remote system automatically on power up. In most cases, the memory-based system is loaded from a remote disk file over a network link—IEEE 802.3 LAN or an HDLC link. A remote download is initiated by auto-boot on power up or by a VCP command. If the VCP interface is a LAN or HDLC interface card, then the VCP commands can be entered from a remote system.

Downloading a memory-based system over an IEEE 802.3 LAN link is described in the *RTE-A System Generation and Installation Manual*. Downloading over an HDLC link is described in the Section “Remote VCP (DSVCP) and Forced Cold Loads” in this manual.

A memory-based system has access to a RAM disk in memory. The RAM disk driver allows a section of memory to be used for file storage in the same manner as a hard disk. This section of memory is called a RAM disk. It allows application programs that use files to run in a memory-based system. Memory limitations should still be considered, however. Refer to the *RTE-A System Manager’s Manual* and *RTE-A System Generation and Installation Manual* for information about the RAM disk.

In a memory-based system, it is important for network managers to understand how to limit or reduce memory usage. Sometimes when reducing memory usage, there may be some limitations when using the NS-ARPA services. The areas to consider when creating a memory-based NS-ARPA system are as follows:

- *NS-ARPA Programs*—Which NS-ARPA programs are absolutely necessary in a memory-based system?
- *Memory Usage*—How do you minimize memory usage?
- *Loading Programs*—How do you load programs into memory from a remote disk?

- *File Usage*—What files can go on the RAM disk? What alternatives are there to error logging to a disk file?
- *Multiuser and Single User Systems*—What are the differences between multiuser and single user systems?
- *Servers, Monitors, and User Programs*—What to do about copies of servers for FTP, NFT, and TELNET?
- *Services*—What happens to some of the NS-ARPA services: TELNET, NFT, FTP, and RPM?

NS-ARPA Programs

NS-ARPA can be tailored to the needs of various systems by including only the programs necessary for the network links and NS-ARPA services being used. Refer to the "Internal Resources" section in this manual to understand which programs are required for each type of link and service. Some suggested examples are as follows:

- If you are not using DS/1000-IV compatible services, then do not include the NS-ARPA programs for these services.
- If you only have LAN links, then you would not need the NS-ARPA programs for Router/1000 links or HDLC links.
- If you transfer files, you can choose one or more file transfer services. If all remote file access will be to another HP 1000, perhaps DS File Transparency is all that may be needed. For file transfers to non-HP systems, use FTP. NFT is used between all HP systems with NS services. RFA is a DS service to access MPE files on HP 3000 systems and FMGR files on HP 1000 systems.

  If a memory-based system uses DS File Transparency to access remote files, then the RTE programs, DSRTR and D.RTR are required. However, if there is no remote file access *into* the memory-based system, then the RTE program, TRFAS, is not needed at the memory-based system.

Minimizing Memory Usage

There are two areas where memory usage can be reduced or minimized:

- network initialization programs
- DSAM

Network Initialization Programs

After NS-ARPA is initialized, the initialization programs are not needed. They would only be needed if you want to shut down or re-initialize NS-ARPA. If you do not, then you can remove the following programs from their memory partitions or purge them from the RAM disk:

```
    NSINIT MMINIT
```

DSAM

The size of DSAM is set in the load file, MMINIT.LOD, by the `EM,150` command. The memory partition used is actually one page larger to accommodate the EMA page table. The “Internal Resources” section of this manual explains how to check for DSAM usage using the NSINF `B` command. NSINF is explained in detail in the *NS-ARPA/1000 Maintenance and Principles of Operation Manual*. The “Internal Resources” section also describes the DSAM requirements of the NS-ARPA services. The size of DSAM is changed by relinking MMINIT. More information is in the “DSAM Reserved Partition Size and Location” subsection of the “NS-ARPA/1000 Program and File Installation” section in this manual.

Loading Programs From a Remote System

There are two ways to load programs from a remote disk into a memory-based system:

- Load programs from the RAM disk.
- Load programs using APLDR.

Loading Programs From the RAM Disk

The program file can be copied to the RAM disk at the remote memory-based system using any remote file copy program such as DS transparency, FTP, NFT, or RFA. Once on the RAM disk, the program can be restored and loaded into memory for execution as on a disk-based system.

The RAM disk must have enough free space to accommodate the file. When the BUILD program initializes a RAM disk LU, it allocates space only for the files it places on the RAM disk leaving no free space. To obtain free RAM disk space, create a RAM disk partition in dynamic memory after boot or purge files that were placed into RAM disk by BUILD.

Loading Programs From APLDR

The program file can be loaded using APLDR. APLDR is a monitor in the DS/1000-IV Compatible Services of NS-ARPA. APLDR is invoked by the REMAT `LO` command or the FLOAD subroutine. These are documented in the *NS-ARPA/1000 DS/1000-IV Compatible Services Reference Manual*. There are a number of restrictions to using APLDR:

- The RTE operating system cannot contain the LOAD and MEMRY modules. These programs also load programs into memory. However, if APLDR is in the system, only APLDR can load programs and manage memory.
- The CI `RP` command can not be used. The RTE `RU` and `XQ` commands for program execution can be used only for programs already loaded into memory. This restriction on program execution also applies to the subroutines, FmpRpProgram and FmpRunProgram. Therefore, all programs scheduled by NS-ARPA software must already be in memory (placed in memory by the RTE BUILD program). NS-ARPA initialization and server programs must also be in memory.
- APLDR does not use dynamic memory. It can only use the fixed memory partitions created by the RTE BUILD program. Because dynamic memory is not available, RAM disk volumes cannot be dynamically allocated or deallocated. All RAM disk volumes must be set up by the RTE BUILD program.
- APLDR can only access program (type 6) files on FMGR volumes. BUILD creates only CI volumes, so APLDR cannot load programs placed on the RAM disk by BUILD.
- CDS programs loaded by APLDR will not share code partitions.

- APLDR and EXECW must be present in the memory-based system where the program is to be loaded. RFAM, the RFA monitor, must be running on the system where the program file exists. The load can be initiated from any node by the REMAT `LO` command or the FLOAD utility.

File Usage

NS-ARPA can operate without most of the files it normally uses. There are alternatives to using hard-disk files:

- *Substitute Logical Unit Numbers*—For NS-ARPA programs requesting an input or output file, specify a logical unit instead. The NSINIT answer file can be read from an LU or from the RAM disk. The NRINIT input file can be accessed remotely with DS transparency, or it can be placed on the RAM disk.

  To specify an LU for event logging, reply to the NSINIT question:

  ```
  * Enter an event log file name [default = /system/ns_event.log]:
  ```

  And enter a `1` for LU 1, the system console, or an LU of a device. Do not put the event log file onto a RAM disk volume. It is a file that grows as errors are logged, and the RAM disk space is limited. Also, the event log file on a RAM disk would not be saved when the system is rebooted (downloaded) or when power to memory is lost.

  You can consider removing EVMON (and event logging) altogether. Unfortunately without logging, NS-ARPA has no way to report errors. To disable logging, do not put EVMON into your memory-based system. Then when NSINIT is run, it prints the following messages:

  ```
  ** (209) NS:  Error SC05 scheduling EVMON (EXEC). **
  ** (200) NS:  WARNING: Required monitor not found. **
  ```

- *Remove Error Message Files*—Removing the error message files results in displaying only error numbers. If NSINIT.MSG is not in the `/SYSTEM` directory, then NSINIT displays only numeric error and warning codes. If NSERRS.MSG is not in the `/SYSTEM` directory, then NFT, TELNET, and FTP display only error numbers. Error numbers and warning codes are explained in the *NS-ARPA/1000 Error Message and Recovery Manual*.

- *Remove Help Files*—Removing the help files deletes the help provided for NS-ARPA commands. NS-ARPA services will still operate.

- *Put /SCRATCH.DIR into the RAM Disk*—FTP uses the `/SCRATCH.DIR` directory for temporary files when it receives directory list commands or commands that use file masks from the FTP user program.

Multiuser and Single User Systems

Most memory-based systems will not have multiuser capabilities. If you are accustomed to multiuser systems, you need to understand the differences between multiuser and single user systems. The VC+ product contains the programs necessary for a multiuser environment and the modules required to run CDS programs. NS-ARPA/1000 requires the CDS modules in order to operate, but not the multiuser programs.

The RTE program, PROMT, determines whether a system is multiuser or not. In a multiuser system, PROMT is normally enabled as the primary program for all terminal LUs. The first time it runs, PROMT initializes the multiuser system. It restores (`RP`) LOGON and CI (calling it CM) if they are not already restored. Therefore, a multiuser system requires PROMT, LOGON, CM, and a `/USERS` directory with the files LOGONPROMPT, MASTERACCOUNT, MASTERGROUP, and any user account files. If these are not present, you have a single user system.

A single user system has all programs running in a single session, the system session. There are no user sessions and no logging on or off. All users are superusers.

In a single user system, there is no file protection security. Thus, anyone on the network can purge or overwrite any file on the single user system.

Servers, Monitors, and User Programs

FTP, NFT, and TELNET all have servers, user programs, and monitors as listed below.

Table C-1. Services, User Programs, Monitors, and Servers

Service	User Program	Monitor	Servers
TELNET	TELNET	INETD	TNSRV
FTP	FTP	INETD	FTPSV
NFT	DSCOPY	NFTMN	CONSM, PRODC, PRDC1

When you initiate an FTP, NFT, or TELNET request, you do so from a user program. The user program establishes the remote connection by communicating with the monitor program at the remote node. The monitor then makes a copy of the server program and passes the connection to it. The user program then communicates with the server to fulfill the user's request. In the meantime, the monitor waits for other connection requests from other user programs. NFT is more complex, because it can operate among three nodes. Refer to the internal details of NFT in the *NS-ARPA/1000 Maintenance and Principles of Operation Manual*.

The INETD program is normally linked as a VMA program with a working set size of 5 pages and a VMA size of 100 pages. In most cases, INETD will only use 2-3 pages of E/VMA. For memory-based systems, HP recommends that you modify INETD.LOD and link INETD as an EMA program with an EMA size of 5 pages. If INETD is `RP`ed into the memory-based system by BUILD, it should not be restarted.

The server programs FTPSV and TNSRV (scheduled by INETD) may exist as type 6 files on the RAM disk or they can be RPed by BUILD. When INETD receives an FTP or TELNET connection, it starts with the name FTPSV and TNSRV and uses the same sequence of names as FmpRpProgram (that is, TNS.A, TSN.B, . . .) to search for a dormant program or an unused name. If INETD finds a dormant program, INETD uses it. If INETD finds an unused name, INETD will RP the server program from a disk file and give it that name.

For NFT, the monitor program, NFTMN, can be loaded into the memory-based system using the RTE-A BUILD program's `RP` command. The NFT server program files must be placed on the RAM disk if multiple copies will be run simultaneously. If only a single copy of each NFT server is needed, they can be RPed into the memory-based system by BUILD.

For each connection initiated, a new copy of the user program is needed. The type 6 or `.RUN` files should be placed in the `/PROGRAMS` directory on your RAM disk. These programs can be copied or cloned and executed from the RAM disk as if they are on a disk-based system. Optionally, multiple copies of the user programs can be `RP`ed by BUILD. However, each copy must have a unique program name. Use one of those unique program names to schedule it. The number of user programs determines how many users can use a service. For example, to allow two TELNET sessions, you can `RP` TELNET as `TELN1` and `TELN2`. These names can then be used to schedule a copy of the TELNET program.

More TELNET Considerations

This subsection explains more considerations for TELNET.

HPMDM

HPMDM does the following three tasks:

- displays messages on the system console for each connection and disconnection
- terminates a user session when TELNET or the modem connection is closed
- closes the connection when a remote user logs off

In a memory-based system without user sessions, the only service that HPMDM provides is displaying messages on the system console. Therefore, if you do not care to see these messages, do not put HPMDM into your system. Or modify HPMDM to fit your needs.

Setting CI as the Primary Program

PROMT is the default primary program scheduled on an interrupt from a terminal or TELNET pseudo terminal LU. PROMT cannot be used in a single-user system. Therefore, the default primary program must be set to another program. The following commands explain how to make a copy of CI as the primary program and CM as the secondary program for a TELNET pseudo terminal LU.

1. Put the following commands in your RTE BUILD command file or in the welcome file to make copies of CI. `RP` a copy of CI for each TELNET pseudo terminal LU giving each a unique name. `RP` one copy of CI with the name of CM.

```
RP,/PROGRAMS/CI.RUN,CI xxx
      .
      .
RP,/PROGRAMS/CI.RUN,CI yyy
RP,/PROGRAMS/CI.RUN,CM
```

If put in the welcome file, then /PROGRAMS/CI.RUN must exist on the RAM disk and your system must be able to load programs from the RAM disk.

2. Use one of the following commands to schedule one of the CI programs as the primary program. If HPMDM is being used, it must be run to enable the primary program. In this example, the TELNET pseudo terminal LU is LU 140.

   ```
   CN,140,20B,CI xxx          In this case, HPMDM is not on the system.
   ```

   or

   ```
   RU,HPMDM,140,PR=CI xxx
   ```

3. Use the CN command to specify CM as the secondary program for each TELNET pseudo terminal LU (the example shows LU 140). CM must already have an ID segment.

   ```
   CN,140,40B,CM
   ```

When a TELNET connection is established, the TELNET server responds with the usual `Welcome to TELNET` message. After you hit carriage return, the pseudo terminal driver schedules the primary program. If CI is the primary program, the CI prompt is displayed.

When you are finished using TELNET, you must explicitly close it. Hit the TELNET escape character (the default is CONTROL]) to get the TELNET prompt. Then enter the TELNET command, CLOSE or QUIT.

More NFT and FTP Considerations

There is no file access security in a single user system. User accounts and passwords are not checked when an NFT or FTP request arrives at a single user system. File access is given to all requests. The server programs for NFT and FTP must execute in the system session. Thus, you are given full access into the single-user system. Anyone on the network can purge or overwrite any file using FTP or NFT.

RPM Considerations

A single user system with RPM can only be a parent node, not a child node. RPM can only create and control processes in a system with user sessions. Therefore, a single user system cannot be an RPM child node. The RPM monitor program, RPMMN, requires multiuser system capabilities and should not be used in a single user system. Programs in a single user system can still call RPM routines to create and control programs at other nodes.

Glossary

address

Used to indicate where nodes are located in a network. Addresses are usually numeric. In NS-ARPA/1000, nodes are assigned different types of addresses. *See also* **Internetwork Protocol address, Router/1000 address**, and **LAN station address**.

ANH

See **Appropriate Next Hop**.

APLDR

The DS/1000-IV Compatible Services absolute program loader. Processes the REMAT `LO`, `IO`, and `PL` commands and FLOAD utility call. (The `LO` command and FLOAD call can be used to load programs into memory-based systems only.)

Application Layer

Layer 7 of the OSI model. Tasks include the user interface to remote services.

Appropriate Next Hop (ANH)

The next node to which IP is to route a message. *See also* **Internetwork Protocol**.

ARP

Translates Internet (IP) addresses to physical addresses. Like Probe, ARP is not directly accessible to users.

ARPA

Advanced Research Projects Agency. ARPA services supported by NS-ARPA/1000 include TELNET (virtual terminal service) and FTP (File Transfer Protocol). *See also* **TELNET** and **FTP**.

asynchronous mode

A mode of data exchange utilized by NetIPC processes. When NetIPC processes exchange data in asynchronous mode, send and receive requests do not cause the calling process to be suspended if a request cannot be immediately satisfied. Instead, a “would block” error is returned and the calling process is free to perform other tasks before retrying the request.

Berkeley Sockets

See **BSD IPC**.

Bisync

A type of communication link used by NS-ARPA/1000 to connect NS-ARPA/1000 to DS/3000 systems. Bisync links support only DS/1000-IV Compatible Services (RTE-MPE), and have no store-and-forward or rerouting capabilities. They can be hardwired or modem connections.

Bisync ID Sequences

Local and remote ID sequences can be assigned to limit access to the HP 1000 if you have an HP 3000 connected to the HP 1000 via a Bisync dial-up link. When the HP 1000 or an HP 3000 attempts to establish communication over the link, each machine sends its local ID sequence, which the other machine compares with its remote ID sequence.

BREVL

The NS-ARPA/1000 program that terminates event logging (EVMON).

broadcast bus network

A network in which nodes are connected by a linear run of cable. Messages are simultaneously transmitted to every node. Typically, the nodes will process only those messages addressed to them, and ignore all other messages. IEEE 802.3 networks are broadcast bus networks. Compare with *point-to-point network*.

BRTRC

The NS-ARPA/1000 program that terminates tracing (NSTRC).

BSD IPC

Berkeley Software Distribution Interprocess (BSD IPC) provides industry standard libraries and tools for interprocess communication on HP, UNIX, and other systems that have BSD IPC.

buffer area

An area of DSAM that is dynamically mapped and contains Message Buffers (MBUFs).

call socket

Used by NetIPC processes to create and connect virtual circuit sockets.

call socket descriptor

A descriptor that refers to a call socket. NetIPC processes obtain call socket descriptors by invoking the NetIPC calls `IPCCreate` or `IPCGet`.

CNSLM

An NS-ARPA/1000 slave monitor that reports MPE TELL and WARN messages. Used by the DS/1000-IV Compatible Services Transport.

communication link

The software and hardware that moves data from the driver and card of one machine to the driver of an adjacent machine. NS-ARPA/1000 supports the following communication links: IEEE 802.3, Ethernet, HDLC, X.25, and Bisync (to NS/3000 or DS/3000 only).

Connect Site Report (CSR)

Provides information on how to reach a given node within an internetwork and how to reach a given NetIPC socket within a node. CSRs are stored in the Socket Registry. *See also* **Socket Registry**.

CONSM

An NS-ARPA/1000 module used by Network File Transfer. CONSM is required at the Consumer node. *See also* **Consumer**.

Consumer

One of the three logical participants in the Three-Node Model utilized by the NS-ARPA/1000 User Service Network File Transfer (NFT). The Consumer is located on the same node as the target file, consumes the source file data and writes it into the target file. *See also* **Three-Node Model**.

Converters

NS-ARPA/1000 monitors that are used to convert message formats for DS/3000 and DS/1000 services. If Converters are needed, they are scheduled by NS-ARPA/1000 and wait for class get completions.

copy descriptor

A parameter used by NFT to describe the source and target file names, nodes names, logons, and any NFT options that should be used when a file is copied using the DSCOPY command.

cross-system

A general term to mean that two different types of computer systems are communicating with one another. For example, cross-system NFT is supported between NS-ARPA/1000 and NS/3000 computer systems as well as between other NS systems. Refer to the *NS Cross-System NFT Reference Manual* for information about NFT. Cross-system NetIPC is also supported between NS-ARPA/1000 and NS/9000 Series 800.

CSR

See **Connect Site Report**.

datagram service

When a datagram service is utilized, the Network Layer (Layer 3 of the OSI architecture) delivers each message separately; no attempt is made to keep messages in order. Therefore, messages may arrive out of order, or not at all. Because there are no "set up," data transfer, or shutdown procedures, each datagram is sent independently and must contain destination information. The HP protocol PXP provides a datagram service. *See also* **PXP**.

Data Link Layer

Layer 2 of the OSI model. Checks for and corrects transmission errors over the physical link. *See also* **Open Systems Interconnection**.

data vector

A structure used by NetIPC calls that can describe several data objects. The description of each object consists of a byte address and a length. The byte address describes where the object is located and the length indicates how much data the object contains. Any kind of data object (arrays, portions of arrays, records, simple variables, etc.) can be described by a data vector.

DCN

See **Directly Connected Network**.

DCN information

Information configured for IP that indicates all the Router/1000 and LAN networks to which the local node belongs. *See also* **Gateway information**, **NGT information**, and **Directly Connected Network**.

Directly Connected Network (DCN)

The local network. *See also* **network**, **Router/1000 network**, and **IEEE 802.3 network**.

Distributed Executive (DEXEC)

A DS/1000-IV Compatible Service that allows you to control I/O devices located at remote HP 1000 nodes. DEXEC calls are the distributed equivalent of local RTE EXEC calls.

Distributed System Available Memory (DSAM)

A memory area specifically reserved for NS-ARPA/1000. Most NS-ARPA/1000 tables are stored in DSAM. NS-ARPA/1000 messages and messages for DS/1000-IV Compatible Services sent over non-Router/1000 links are stored in DSAM before transmission. DSAM is controlled by the NS-ARPA/1000 Memory Manager, and is implemented as a Sharable Extended Memory Access (SHEMA) partition. *See also* **Memory Manager**, **Message Buffers**, **clusters**, and **buffer area**.

DLIST

The NS-ARPA/1000 remote directory list monitor. Lists contents of FMGR directories. Used by the DS/1000-IV Compatible Services.

DSCOPY

An NS-ARPA/1000 module used by Network File Transfer. Establishes a VC connection with NFTMN at the Producer node. *See also* **Producer** and **NFTMN**.

DS File Transparency

A feature of the RTE operating system that allows you to access remote files. Also called Transparent File Access (TRFAS).

DSLIN

The NS-ARPA/1000 module used to establish PSI BISYNC connections to HP 3000s.

DSMOD

An NS-ARPA/1000 program that allows you to alter parameters for DS/1000-IV Compatible Services that are set by NSINIT during initialization. DSMOD allows the user to change the HP 3000 ID sequence, re-enable a link, display the Nodal Routing Vector (NRV), change the non-session password, schedule additional monitors, adjust timing, change the default session user name, and change the NRV.

DS/1000-IV Compatible Links

See **HDLC**, **Bisync**, and **X.25**.

DS/1000-IV Compatible Services (RTE-MPE)

A term used to describe services that can be used for backward compatibility with DS/3000 or NS/3000 nodes over Bisync or X.25 links. These services include Program-to-Program Communication (PTOP), Remote File Access (RFA), RMOTE, and the Utility Subroutines.

DS/1000-IV Compatible Services (RTE-RTE)

A term used to describe services that can be used for backward compatibility with DS/1000-IV nodes. These services include Distributed Executive (DEXEC), Program-to-Program Communication (PTOP), REMAT, Remote Database Access (RDBA), Remote File Access (RFA), Transparent File Access (TRFAS), Utility Subroutines, Remote I/O Mapping, Remote System Download, and Remote Virtual Control Panel (VCP).

DS/1000-IV Compatible Transport

Transport used by the DS/1000-IV Compatible Services (RTE-RTE) and (RTE-MPE) and DS/1000-IV Compatible Links that are configured with RTR LIs. Each of these transports is responsible for delivery of data between the source and the destination. *See also* **Router Link Interface**.

DSRTR

The NS-ARPA/1000 transparent file access master. DSRTR is part of the RTE operating system.

DSTES

An NS-ARPA/1000 module that verifies the PTOP software to a DS/3000 node.

DSVCP

The DS Virtual Control Panel operator interface module for remote control of the A/L-Series front panel. DSCVP is a DS/1000-IV Compatible Service. Users must run the program DSVCP to access the control panel of a slave computer, and consequently access and alter its memory, and CPU and I/O registers. DSVCP can access the boot loader programs in the VCP and cause various programs to be downloaded, via the NS-ARPA link, or another loading device. DSVCP can be used over HDLC RTR links only.

Dynamic Rerouting

The capability to automatically reroute messages around inoperative HDLC links without user intervention. Dynamic rerouting is an option provided by the Router/1000 protocol.

Ethernet Local Area Network

Ethernet is a de-facto standard link level protocol. Ethernet defines a baseband, coaxial, bus media, and the Media Access Method CSMA/CD. IEEE 802.3 and Ethernet nodes can coexist on the same cable, but cannot communicate with each other.

event messages

Messages sent between protocol modules to indicate events and pass references to path records.

EVMON

The NS-ARPA/1000 Event Monitor. Logs disasters, errors, warnings, and internal state information.

exception select

Can be performed by using the NetIPC call `IPCSelect.` NetIPC processes can determine whether certain connections have been aborted by performing an exception select.

EXECM

The NS-ARPA/1000 remote EXEC slave monitor. Services remote EXEC (DEXEC) calls. These requests may come from other HP 1000 or HP 3000 nodes. Used by the DS/1000-IV Compatible Services. *See also* **Distributed Executive**.

EXECW

The NS-ARPA/1000 remote “schedule with wait” monitor. Services remote EXEC (DEXEC) requests to schedule programs with wait. To run a program from REMAT or to execute the `LO` (load) and `PL` (program list) commands on memory based nodes, the remote node must have EXECW. Used by the DS/1000-IV Compatible Services.

FCL66

The NS-ARPA/1000 utility that is used to programmatically force cold load slave computers over an HDLC RTR link.

FMTRC

The NS-ARPA/1000 program that formats trace files produced by NSTRC. *See also* **NSTRC**.

FSRV

FSRV is the HP 1000 file server program that services file access requests from remote NFS clients. See **NFS**.

FTP

File Transfer Protocol is an ARPA service that allows you to copy files from one node to another. The other computer must also support FTP.

fully-qualified node name

An NS-ARPA/1000 node name that includes all three parts of the syntax (i.e., the node, domain, and organization). *See also* **node name**.

gateway node

A node that is a member of two or more networks and allows communication between the networks to which it belongs.

Gateway Table (GT)

A table used by the Internetwork Protocol (IP) to determine which gateway to route through to reach a remote network. *See also* **Internetwork Protocol** and **DCN information**.

global area

An area in DSAM that is always mapped in. Memory Manager stores information in the global area that it needs to access frequently or quickly. See **DSAM** and **Memory Manager**.

GRPM

The NS-ARPA/1000 transport monitor that acts as the RTE-RTE request/reply processor. GRPM is scheduled by ID*66 to allocate a class buffer in SAM to receive incoming messages. GRPM routes incoming messages to the Slave Monitor's class numbers.

guardian node

A node that allows 91750 nodes to communicate with nodes on remote networks. Guardian nodes remove IP headers from messages sent over non-Router/1000 links before delivering them to the 91750 node. Guardian nodes also add IP headers to messages sent from 91750 nodes that are to be forwarded to remote networks.

HDLC

A type of communication link used by NS-ARPA/1000 to connect HP 1000s. Can have a RTR LI. Networks composed of HDLC links are point-to-point networks and can have arbitrary topologies. HDLC links can be hardwired or modem connections. *See also* **point-to-point network**.

hierarchical

A point-to-point network topology. The hierarchical topology is sometimes used with supervisory-control applications, where large databases exist at one node, possibly along with control programs, that are accessed by nodes lower in the hierarchy. You can also use hierarchical topologies for distributed database applications.

IEEE 802.3 Link Interface

NS-ARPA/1000 link interface type that supports a single communication link type: IEEE 802.3. The LI software determines which IEEE 802.3 communication LU to use and sends the message to the IEEE 802.3 driver. At this point the control of the message moves from NS-ARPA/1000 to LAN/1000. *See also* **communication link** and **Link Interface**.

IEEE 802.3 Local Area Network Link

A type of communication link used by NS-ARPA/1000 to join HP 1000s over a relatively small geographical area to form a LAN. Provides fast links and requires less hardware than point-to-point links for networks with several nodes. IEEE 802.3 networks are broadcast bus networks. IEEE 802.3 and Ethernet nodes can coexist on the same cable, but cannot communicate with each other. *See also* **broadcast bus network** and **Ethernet**.

IEEE 802.3 network

A group of nodes connected to the same LAN bus. *See also* **broadcast bus network**.

IEEE 802.3 protocol

A protocol used at the Subnet or Intranet Layer (Layer 3s of the NS-ARPA/1000 architecture). IEEE 802.3 defines some layer 3s functions for messages sent over IEEE 802.3 links. With IEEE 802.3 LAN networks, messages are transmitted to every node in the network and a node accepts only those messages that are addressed to it.

IFPM

An NS-ARPA/1000 transport monitor that acts as an interface between DS/1000-IV Compatible Services and IEEE 802.3 links for outbound messages. Implements the Interface Protocol (IFP).

Inbound Address List

A list of LAN addresses maintained by each node in a LAN network. A given node will receive only those messages that have a destination address that matches an address on its Inbound Address List. Each node's Inbound Address List contains the node's station address. In addition, a group of nodes may have the same multicast address in their Inbound Address List. *See also* **multicast address**.

Inbound Proxy Address

Used by LAN nodes to receive Proxy Name Requests. The Inbound Proxy Address is added to the Inbound Address List of Probe Proxy Servers only.

Inbound Target Address

Added to every LAN node's Inbound Address List at initialization time. The Inbound Target Address allows the LAN nodes to receive Probe Name Requests.

INETD

Internet Network Services Daemon that monitors all incoming connection requests from TELNET, FTP, SMTP (Mail/1000), and Remote Print Spooling.

Initiator

One of the three logical participants in the Three-Node Model utilized by the NS-ARPA/1000 User Service Network File Transfer (NFT). The Initiator is located on the system where the copy request originates, receives the user request, and initiates the copy process. *See also* **Three-Node Model**.

INPRO

The inbound message processor for the NS-ARPA/1000 transport. Contains the inbound NS-ARPA protocol modules.

Interchange Format

One of the file copying formats used by the NS-ARPA/1000 User Service Network File Transfer (NFT). Interchange Format must be invoked explicitly using one or more of NFT's Interchange Format options. Interchange Format is useful when you want to change certain source file attributes, such as record length, in the target file.

Interface Protocol (IFP)

Provides the interface between NS Common Services Transport and DS/1000-IV Compatible Services Transport.

internetwork

Several networks that are joined, or concatenated, to form a network of networks.

internetwork communication

Communication between networks.

Internetwork Protocol (IP)

The NS-ARPA/1000 protocol based on the Defense Advanced Research Projects Agency's (DARPA) standard that is implemented at the Internet Layer (Layer 3i) of the NS-ARPA/1000 architecture. IP is primarily used to route messages between networks via gateways. It provides gateway-to-gateway routing, store-and-forward service between gateways, and message fragmentation and reassembly between source and destination networks. IP routines are contained in the INPRO and OUTPRO message processors. *See also* **INPRO** and **OUTPRO**.

Internetwork Protocol (IP) address

An address used by the NS-ARPA/1000 Services and IP. An IP address consists of two parts: a network address, which identifies the network; and a node address, which identifies the node within a network. A network address is concatenated with a node address to form the IP address and uniquely identify a node within a network within an internetwork.

Intranet Layer

In NS-ARPA/1000, this is layer 3s. Layer 3s is part of the OSI Network Layer and handles intranetwork routing (routing within a network). NS-ARPA/1000 supports four Layer 3s protocols: Router/1000, IEEE 802.3, Ethernet, and X.25. *See also* **Open Systems Interconnection**.

IOMAP

The NS-ARPA/1000 module that provides the user interface for Remote I/O mapping. *See also* **Remote I/O Mapping**.

IP

See **Internetwork Protocol**.

LAN

Local Area Network. *See also* **IEEE 802.3 Local Area Network Link** and **Ethernet Local Area Network**.

LAN station address

An address that HP assigns to each LAN interface card during manufacturing. The station address is used for addressing within the LAN (subnetwork addressing).

layers

Refers to the layers in the seven-layer Open Systems Interconnection (OSI) network architecture model developed by the International Standards Organization (ISO). In the NS-ARPA/1000 network architecture, different transmission and communications tasks are assigned to each layer, which is a logically distinct module. *See also* **Open Systems Interconnection**.

LI

See **Link Interface**.

link

See **communication link**.

Link Interface (LI)

The software that determines which communication link type to use and then passes the message to the appropriate driver for the specific communication link. Communication links are bound to link interface types at NS-ARPA/1000 initialization time. NS-ARPA/1000 supports four LI types: Router (RTR), IEEE 802.3 (802), Ethernet (ETHER), and if both IEEE 802.3 and Ethernet are on the same local area network, then the LI type is declared LAN. *See also* **Router Link Interface**, **IEEE 802.3 Link Interface**, **Ethernet Local Area Network**, and **communication link**.

link type

See **communication link**.

local network

The network to which the local node belongs. Also referred to as the Directly Connected Network (DCN).

local node

Refers to the node where you are physically located and logged on, and at which you enter commands.

LOG3K

The NS-ARPA/1000 module that provides operator control over recording of DS messages to and from HP 3000s.

LUMAP

The NS-ARPA/1000 DEXEC request module for Remote I/O Mapping. *See also* **Remote I/O Mapping**.

LUQUE

The NS-ARPA/1000 module that provides class buffers for Remote I/O mapped data transfer. *See also* **Remote I/O Mapping**.

#MAST

Handles request buffers for master routine subroutine calls. Adds a Router/1000 header to the request and passes it to GRPM or to the transmission LU. Used by the DS/1000-IV Compatible Services.

master-slave protocol

The sequence of messages exchanged between master and slave Program-to-Program (PTOP) programs. *See also* **Program-to-Program Communication**.

MATIC

An NS-ARPA/1000 module that provides time-out processing for Message Accounting. Used by the DS/1000-IV Compatible Services and Links.

Memory Manager

Manages DSAM, the NS-ARPA/1000 memory area. Memory Manager divides DSAM into three areas: global area, tables area, and buffer area. *See also* **global area**, **tables area**, and **buffer area**.

Message Accounting (MA)

A datagram-oriented protocol that retransmits lost messages and suppresses duplicate messages. The DS/1000-IV Compatible Transport (RTE-RTE) uses Message Accounting.

Message Buffers (MBUFs)

Part of the DSAM buffer area. An MBUF is 1 page (2048 bytes) of physical memory.

monitor

In NS-ARPA/1000, a monitor is a software module that is scheduled at node initialization time, or by DSMOD, and remains scheduled until NS-ARPA is shut down. There are three types of NS-ARPA/1000 monitors: Transport Monitors, Slave Monitors, and Converters. *See also* **Transport Monitors**, **Slave Monitors**, **Converters**, and **watch dogs**.

MMINIT

The Memory Manager initialization program. MMINIT sets up DSAM. MMINIT is scheduled by NSINIT at initialization. *See also* **Memory Manager**, **DSAM**, and **NSINIT**.

multicast address

An address that may be included in the Inbound Address List of a node in a LAN network. A group of nodes may have the same multicast address in their Inbound Address List. Messages with that multicast address are received by all nodes in that group. Probe uses multicast addresses to send messages to HP nodes in a LAN. Two Probe multicast addresses are used at each LAN node: a Target Address and a Proxy Address. *See also* **Inbound Address List**, **Probe**, **Target Address**, and **Proxy Address**.

MVCP3

The NS-ARPA/1000 module used to install the PTOP slave program `COPY3K.PUB.SYS` on an HP 3000 for use in implementing the RMOTE `MO` command. *See also* **RMOTE**.

name

Used to identify objects, such as nodes and sockets. In NS-ARPA/1000, nodes are assigned node names. *See also* **node names**.

NetIPC

See **Network Interprocess Communication**.

NetIPC root socket

NetIPC allocates one root socket for each NetIPC process. The root socket is used as an endpoint to lower level protocols and sets up a path for any call or VC sockets requested by the user.

NetIPC user record

NetIPC allocates one user record for each NetIPC process. NetIPC uses this record to keep track of that process's NetIPC sockets and other resources used for NetIPC.

network

A group of computer systems connected so that they can exchange information and share resources. More specifically, see **Router/1000 network** and **IEEE 802.3 network**.

network architecture

A structured, modular design for networks.

network boundary

The division between networks in an internetwork. *See also* **internetwork**.

Network File Transfer (NFT)

An NS-ARPA/1000 User Service that allows you to copy files from one node to another interactively or programmatically. Cross-system NFT is also supported.

Network Interprocess Communication (NetIPC)

An NS-ARPA/1000 User Service that allows autonomous processes running concurrently at different nodes to exchange information in a peer-to-peer manner. Cross-system NetIPC is also supported.

Network Layer

Layer 3 of the OSI model. Tasks include determining the routes messages take to get from one node to another. In NS-ARPA/1000, this layer is split into two sub-layers, 3i and 3s. Layer 3i handles internetwork routing and 3s handles subnetwork or intranetwork routing, which is routing within the network.

network map

A diagram of the links and nodes in your network. A network map should include node information (system type and resources, peripherals, amount of memory, services supported) and link information (location of coaxial cables, terminators, repeaters, AUI cables, and MAUs). In addition, you should mark network boundaries, link interface types, card types, and the names and addresses that you assign.

For an internetwork, the network map shows how different networks are connected. A network map for an internetwork includes the following: the network names and types (and, if applicable, a unique IP address for each network), the gateway nodes, and the network boundaries.

Network security code

A code consisting of two alphanumeric characters that is required for the following tasks: shutting down NS-ARPA/1000 (via NSINIT); establishing Remote I/O maps; modifying timing parameters; changing the Nodal Routing Vector (via DSMOD).

Network User's security code

A code consisting of two alphanumeric characters that is required to execute the REMAT `SW` (switch) command; thus it is used to restrict REMAT access to remote nodes. *See also* **REMAT**.

NFS

The Network File System (NFS) is a protocol developed by Sun Microsystems that allows nodes in a computer network to access files at other nodes. (NFS is a trademark of Sun Microsystems, Inc.)

NFT

See **Network File Transfer**.

NFTMN

The Network File Transfer monitor program. NFTMN is scheduled at initialization time and remains scheduled, waiting for an `IPCRecvCn` call, until NS-ARPA is shut down. At the Producer node, NFTMN schedules PRODC; the NFTMN at the Consumer node schedules CONSM. *See also* **Producer**, **Consumer**, **PRODC**, and **CONSM**.

Nodal Path Reports (NPRs)

Contains information stored in DSAM by the Nodal Registry software. Nodal Path Reports map node names to address information. NPRs are indexed according to node name, and each contain an IP address (or addresses, if the node belongs to multiple networks). If the node has LAN links, the NPR can also contain a LAN station address for each link. In addition, an NPR denotes which protocols and NS Common Services the node supports.

Nodal Registry

The NS-ARPA/1000 software that manages information that the transport and services use to establish connections with remote nodes.

Nodal Routing Vector (NRV)

Used by Router/1000 for subnet routing with Router/1000 networks and for address information for DS/1000-IV Compatible Services (RTE-RTE).

node

A computer system in a network.

node names

Each computer system, or node, in an NS-ARPA/1000 network has a node name. NS-ARPA/1000 node names contain three fields: a node, domain, and organization. *See also* **fully-qualified node name**.

node number

See **Router/1000 address**.

NRINIT

The Nodal Registry initialization program.

NRLIST

The Nodal Registry List program. Lists the contents of the Nodal Registry.

NRV

See **Nodal Routing Vector**.

NS Common Services

A term used to refer to the services: Network File Transfer (NFT), Network Interprocess Communication (NetIPC), and Remote Process Management (RPM).

NSERRS.MSG

The name of the Network File Transfer message file. Must be on `/system`.

NSINF

The NS-ARPA/1000 information utility. Prints local address information, configured NS-ARPA/1000 resources, NS-ARPA/1000 network management utilities status and statistics, Memory Manager statistics, NS-ARPA/1000 program information, DS/1000-IV Compatible Services information, Nodal Routing Vector information, Remote Session information, and Message Accounting information.

NSINIT

NSINIT and its subordinate programs (NSPR1, NSPR2, NSPR3, NSPR4, NSPR5, NSPR6, and NSPARS) initialize and shut down the network.

NSINIT.MSG

The name of the NSINIT message file. Must be on `/system`.

NSTRC

The NS-ARPA/1000 trace utility. Records messages at the socket and network level.

Open Systems Interconnection (OSI)

A seven-layer network architecture model developed by the International Standards Organization (ISO). In the OSI model, transmission and communication tasks are assigned to logically distinct modules called layers. Each layer communicates with the layer directly above and below it, and through the layers below it to its peer in the remote computer. The OSI model defines seven layers: Application Layer, Presentation Layer, Transport Layer, Network Layer, Data Link Layer, and Physical Layer.

OPERM

An NS-ARPA/1000 slave monitor that provides remote RTE operator command capability.

OSI

See **Open Systems Interconnection**.

Outbound Proxy Address

Used by LAN nodes to send Proxy Name Requests.

Outbound Target Address

Used by LAN nodes to send Probe name and address requests (Name Requests and Virtual Address Requests).

OUTPRO

The outbound message processor for the NS-ARPA/1000 transport. Contains the outbound NS-ARPA protocol modules.

Packet Exchange Protocol (PXP)

An HP protocol that provides a low-overhead datagram service. PXP is a low-overhead request/reply protocol that is suited for querying data sources. PXP suppresses duplicate replies to a request but does not suppress responses to duplicate requests. PXP retransmits messages that are not acknowledged within a timeout interval. Socket Registry is the only NS-ARPA/1000 service that uses PXP; it is not user accessible. PXP is included in INPRO and OUTPRO.

path

The course within a machine that a message takes, typically through software protocol handlers.

path records

The data structures in which the NS-ARPA/1000 protocols keep address and other context information. The protocol modules also use path records to guide messages to the next appropriate protocol. Path records are logical records; the actual data structures that hold path record information are given different names by the protocols and usually contain other, protocol-specific, information.

path report

Created by the socket registry. Path reports are returned to NetIPC processes in the form of a path report descriptor. The path report contains addressing information that is used by the calling process to direct requests to a certain call socket at a certain node.

path report descriptor

A descriptor used by NetIPC processes to refer to a path report. A NetIPC process obtains a path report descriptor by invoking either the `IPCLookUp` or `IPCGet` call. *See also* **path report**.

PCB

See **Protocol Control Block**, **Probe Control Block**, or **PTOP Control Block**.

Physical Layer

Layer 1 of the OSI model. Transmits the electrical signals over the link.

point-to-point network

A network in which communication travels from one node (point) to another node over the links.

pool LU

Used for X.25 connections to HP 3000s (DS/1000-IV Compatible Services).

POOL Table

Contains entries for each concurrent remote user or program, including monitors, that access the local node to use DS/1000-IV Compatible Services (RTE-RTE).

PRDC1

An NS-ARPA/1000 module used by Network File Transfer. PRDC1 is scheduled by PRODC if the user specifies a file mask. PRDC1 is required at the Producer node if file masks are used. *See also* **PRODC** and **Producer**.

Presentation Layer

Layer 6 of the OSI architecture. Tasks include manipulation of user data such as text compression and encryption.

Probe

A protocol that allows LAN nodes to query one another for Nodal Path Reports and other address information. The Probe protocol provides the following features: node name to Nodal Path Report mapping; IP address to LAN station address mapping; Nodal Registry updates. *See also* **Nodal Path Reports**.

Probe addresses

See **Inbound Address List**, **multicast address**, **Target Address**, and **Proxy Address**.

Probe Control Block (PCB)

Required for each active Probe query.

Probe Proxy Server

Provides Nodal Path Reports (NPRs) for any node in an internetwork. If a LAN has a Probe Proxy Server, all nodes on that LAN can get the NPRs they need from the Probe Proxy Server; at the other nodes on the LAN, only the NPR for that node must be configured. *See also* **Nodal Path Reports**.

Process Number List (PNL)

List containing TCBs used for transactions to HP 3000s (DS/1000-IV Compatible Services). *See also* **Transaction Control Block**.

PRODC

An NS-ARPA/1000 module used by Network File Transfer. PRODC establishes a VC connection with the monitor NFTMN at the Consumer node (the node to which the file will be copied). *See also* **NFTMN**, **Consumer**, **PRDC1**, and **Producer**.

Producer

One of the three logical participants in the Three-Node Model utilized by the NS-ARPA/1000 User Service Network File Transfer (NFT). The Producer is located on the same node as the source file, accesses that file and produces the data that is to be copied.

PROGL

The NS-ARPA/1000 slave monitor used for remote download from Communication Bootstrap Loader (CBL). Can simultaneously handle requests from up to 20 nodes at the same time. User supplied subroutines enable store-and-forward and/or LU to file conversion capabilities. Can be used over HDLC RTR links only. PROGL is used by the DS/1000-IV Compatible Services.

Program-to-Program Communication (PTOP)

An NS-ARPA/1000 User Service that enables a “master” program on one node to exchange information with and control the execution of a “slave” program on another node.

protocol

A set of rules for a particular communication task. A protocol handler or protocol module is a piece of software that implements a particular protocol.

Protocol Control Block (PCB)

Used by TCP and PXP to keep track of protocol-specific parameters such as segment sizes and time values.

protocol EMAs

A term used to describe EMA partitions that are used by the NSINIT subordinate programs (NSPR1, NSPR2, NSPR3, NSPR4, NSPR5, and NSPR6). Protocol EMAs are used to build protocol tables before moving the tables to DSAM.

protocol module

A group of software modules that implement a given protocol. For example, the TCP protocol module is the collective term for the group of software routines that implement the TCP protocol.

Proxy Address

One of the two Probe multicast addresses that must be configured at each LAN node. Probe uses this address for two other addresses: an Inbound Proxy Address and an Outbound Proxy Address. *See also* **Inbound Proxy Address** and **Outbound Proxy Address**.

PTOP Control Block (PCB)

Parameter used in PTOP calls which serves as a control for the data link. *See also* **Program-to-Program Communication**.

PTOPM

The NS-ARPA/1000 PTOP communication slave monitor. Handles programmatic `POPEN`, `PREAD`, `PWRIT`, `PCONT`, and `FINIS` requests and REMAT commands `SO` (slave off) and `SL` (slave list) on the slave side.

PXP

See **Packet Exchange Protocol**.

QCLM

An NS-ARPA/1000 transport monitor that acts as a communications error logger. QCLM prints most errors for the DS/1000-IV Compatible Services modules, the DS/1000-IV Compatible Transport, and the HDLC driver.

QUEUE

An NS-ARPA/1000 program that is scheduled by the interface driver to allocate a class buffer in SAM to receive incoming messages.

QUEX

An NS-ARPA/1000 transport monitor used for HP 3000 communication.

QUEZ

An NS-ARPA/1000 transport monitor used for HP 3000 I/O completions.

raw NPRs

Character strings that the Nodal Registry initialization program, NRINIT, uses to build Nodal Path Reports. A raw NPR contains a node name and an IP address or addresses. If the node has a LAN link, the raw NPR can also contain a LAN station address or addresses. *See also* **Nodal Path Reports**.

RDBAM

The NS-ARPA/1000 remote database access slave monitor.

READR

A LAN/1000 module for the Node Manager.

read select

A read select can be performed by using the NetIPC `IPCSelect` call. NetIPC processes can determine whether certain VC sockets are readable by performing a read select. A readable socket is one that can immediately satisfy a receive request for a number of bytes greater than or equal to its read threshold.

read threshold

Used by NetIPC processes that exchange data in asynchronous mode. A NetIPC process determines whether a VC socket is readable by examining the socket's read threshold. A VC socket is considered readable if it can immediately satisfy a receive request for a number of bytes greater than or equal to its read threshold.

redundant links

Provide alternative routes between nodes in a Router/1000 network. If a node or link fails, the nodes can still communicate by rerouting around the failure. In a Router/1000 network with Dynamic Rerouting, the rerouting is automatic; otherwise you must explicitly configure new routes around the failure. *See also* **Dynamic Rerouting**.

REMAT

A DS/1000-IV Compatible Service that allows you to send RTE commands, or special REMAT commands, to any HP 1000 computer in your network.

Remote Database Access

A DS/1000-IV Compatible Service that allows you to access an IMAGE database at a remote HP 1000.

Remote File Access (RFA)

A DS/1000-IV Compatible Service that enables you to perform I/O operations to files located at remote nodes.

Remote I/O Mapping

A DS/1000-IV Compatible Service that maps I/O requests from one node to another, allowing resource sharing.

remote network

Any network in the internetwork to which the local node does not belong.

remote node

Refers to a node that is not physically located where you are, and which you communicate with via data communication.

Remote Process Management (RPM)

An NS-ARPA/1000 User Service that allows you to schedule, control, or terminate programs located at the same or different HP 1000 nodes in your network.

Remote System Download

A DS/1000-IV Compatible Service that allows you to download an operating system file to a remote HP 1000.

Remote Virtual Control Panel (VCP)

A DS/1000-IV Compatible Service that allows you to operate the VCP of a remote node. This feature is useful for applications that require a remote, terminal-less node (such as a harsh operating environment).

rerouting

The capability to reroute messages around inoperative links.

resource sharing

The most significant feature of a network. Elements at each node are accessible from other nodes in the network. These elements may include disk files, printers, magnetic tapes, terminals, and other programs.

RFA

See **Remote File Access**.

RFAM

The NS-ARPA/1000 remote file access slave monitor. Used for FMGR files only and must reside at the node at which the file resides. Used by the DS/1000-IV Compatible Services.

ring

A point-to-point network topology. The ring is a string topology with an additional link between the end nodes. The store-and-forward delay is half that of a string topology because the maximum number of intervening nodes is halved. The ring topology is suited for data-sharing applications in which data stored at various nodes are accessible from all nodes. Ring networks are less vulnerable than string networks. If any one link fails, all the nodes can still communicate by rerouting around the failure. *See also* **string**.

RMOTE

A DS/1000-IV Compatible Service that creates an interactive session for you on a remote HP 3000 in your network, making your terminal appear to be directly connected to the other system.

root socket

See **NetIPC root socket**.

route

The course through the network that a message takes from a source node to a destination node. A route can pass through intervening nodes.

Router Link Interface (RTR LI)

An NS-ARPA/1000 Link Interface that supports three communication link types: HDLC, X.25, and Data Link (master only). RTR LIs offer the following features: Dynamic Rerouting (HDLC only), low-overhead for DS/1000-IV Compatible Services, high-overhead for NS Common Services. The nodes linked by the RTR LI are members of the same network. *See also* **Dynamic Rerouting** and **communication link**.

Router/1000

A protocol used at the Subnet or Intranet Layer (Layer 3s of the NS-ARPA/1000 architecture). Router/1000 provides store-and-forward and Dynamic Rerouting services for messages sent over HDLC links. Optionally provides rerouting and Message Accounting. *See also* **rerouting** and **Message Accounting**.

Router/1000 address

An address used by the DS/1000-IV Compatible Transport and Router/1000 software. Called “node number” on 91750 nodes. Must be unique to the internetwork.

Router/1000 header

Message header used for DS/1000-IV Compatible Services and Transport. In the 91750 product, this header is referred to as the DS message header.

Router/1000 network

A group of nodes that are connected by RTR LIs. There may be redundant, non-RTR LIs in a Router/1000 network, but for any two nodes in the network, a route must exist between them that consists only of RTR LIs.

RPCNV

The NS-ARPA/1000 to DS/3000 reply converter.

RPM

See **Remote Process Management**.

RQCNV

The NS-ARPA/1000 to DS/3000 request converter.

RTR LI

See **Router Link Interface**.

#SEND

An NS-ARPA/1000 module that is used with Dynamic Message Rerouting. #SEND sends update messages to neighboring nodes. #SEND is part of Router/1000.

#SLAV

Called by the DS/1000-IV slave monitors to send a reply and data, if any, back to the origin node. Used by the DS/1000-IV Compatible Services.

SAM

See **System Available Memory**.

SAP

Service Access Point. The only IEEE-defined SAP currently defined is 6 for IP (Internetwork Protocol).

SBUFs

See **socket buffers**.

Session Layer

Layer 5 of the OSI model. Tasks include connection establishment negotiation at remote nodes.

Slave Monitors

NS-ARPA/1000 monitors that service incoming requests for local resources from remote nodes.

SMB

See **System Memory Block**.

socket

Used to establish communication between NetIPC processes. Processes make use of sockets via the NetIPC calls to establish connections and exchange data. The NS-ARPA/1000 Transport Layer's Transmission Control Protocol (TCP) regulates the transmission of data to and from sockets.

socket buffers

Memory Manager allocates an inbound and an outbound SBUF for each NetIPC socket to hold queued inbound and outbound messages. *See also* **Memory Manager**.

Socket Registry

Contains a listing of all the named call sockets that reside at a node. NetIPC processes reference call sockets created by other processes by passing a socket name and the corresponding node name to the socket registry software. The socket registry determines which socket is associated with the name and formats the address information to that socket into a path report which it returns to the inquiring process. Socket Registry is not directly accessible by the user.

star

A point-to-point network topology. The star topology is often used for centralized data collection, supervisory control, or in an application where the outlying nodes have little storage capacity. It is also used when a central node has a large database or control program that is accessed by the other nodes. In a star network, there is at most one intervening node between any two nodes. Star networks are vulnerable to failure of the central node. If the central node fails, no network communication is possible.

station address

See **LAN station address**.

store-and-forward

A method of forwarding messages in a network. In a store-and-forward network, a node can send a message to another node to which there is no direct link. Intermediate nodes can forward the message to the correct destination node. Messages can be stored and forwarded between several nodes.

stream mode

The mode of data transfer used by NetIPC processes. Data transferred between two NetIPC processes is treated as byte stream. When data arrives at a destination VC socket, it is simply appended to any data that may have already been sent to that socket. No attempt is made to preserve boundaries between data sent at different times.

string

A point-to-point network topology. The string topology requires one less communication link than there are computers in the network, and requires the fewest number of links. For communication between non-adjacent nodes, messages are stored and forwarded by intervening nodes. If a link fails, the nodes separated by the failure will not be able to communicate.

subnet mask

A mask that specifies the subnet number in an IP address. The node address portion of an IP address is divided into a subnet number and a node number. The bits in the subnet mask are set to 0 for the node number portion and 1 for the network address and subnet number portions.

subnetting

An optional addressing scheme that partitions the node address portion of an IP address into distinct subnetworks. The node address is divided into a subnet number and a node number. Subnetting allows you to use one network address for two or more physically distinct networks. Each network is then a subnetwork. The physical networks are connected via gateways.

subnetwork

A network that may be a member of an internetwork. *See also* **Intranet Layer** and compare with **Internetwork Protocol** and **internetwork**.

subordinate programs

Programs that are part of NSINIT. *See also* **NSINIT**.

synchronous mode

A mode of data exchange utilized by NetIPC processes. When NetIPC processes exchange data in synchronous mode, send and receive requests cause the calling process to be suspended until the request can be satisfied, a synchronous time-out occurs, or an error is detected.

SYSAT

The NS-ARPA/1000 System Attention Module required for Remote I/O Mapping. Sends a message to a remote system to set a program's break flag. Also used to send the System Attention request to a remote system. Used by the DS/1000-IV Compatible Services.

System Available Memory (SAM)

In NS-ARPA/1000, SAM is used as a buffer area between the NS-ARPA/1000 device drivers and the Transport. All inbound and outbound NS-ARPA messages pass through SAM.

System Memory Block (SMB)

A memory area in the system map that is specified at system generation time by the generator `MB` command. In NS-ARPA/1000, SMB is used for some tables provided for DS/1000-IV compatibility.

tables area

An area in DSAM that is dynamically mapped in as tables are requested. The tables area contains the dynamically-sized NS-ARPA tables used by protocols and services, which are referenced in the global area. *See also* **DSAM**.

TCB

See **Transmission Control Block**.

TCP

See **Transmission Control Protocol**.

TELNET

An NS-ARPA/1000 User Service that allows you to have a virtual terminal connection to another computer in your network. The other computer must also support TELNET. TELNET stands for TELecommunications NETwork. TELNET communicates using the TELNET protocol which is a standard ARPA service.

Three-Node Model

The model utilized by the NS-ARPA/1000 User Service Network File Transfer (NFT). The Three-Node Model has three logical participants: the Initiator, the Producer, and the Consumer. All three participants are logically distinct. They may be three separate processes on three separate nodes, or any two, or all three, may reside on the same node. *See also* **Initiator**, **Producer**, and **Consumer**.

Transaction Status Table (TST)

Used to keep track of all master requests from an HP 3000. Provides storage for information from the DS/3000 fixed format header and DS/1000-IV information generated by RQCNV.

Transmission Control Block (TCB)

A TCB is allocated for each program that uses DS/1000-IV Common Services. The TCB keeps track of requests until a reply is received or the request times out.

Transmission Control Protocol (TCP)

The Transport Layer (Layer 4) of the NS-ARPA/1000 architecture implements the Transmission Control Protocol (TCP), which is based on the DARPA standard. TCP is a stream-based (rather than a message-based) protocol that provides non-duplicated, in-sequence data delivery. TCP accepts arbitrarily long data buffers, segments them into packets, and sends each packet separately. TCP keeps track of the bytes sent and retransmits them if they are not acknowledged within a timeout interval. TCP at the receiving node reassembles the packets, so that they are delivered to the user (NetIPC) in order (in-sequence delivery). All NS-ARPA/1000 services use TCP except Socket Registry and the DS/1000-IV Compatible Services. TCP is part of INPRO and OUTPRO.

Transparent File Access (TRFAS)

A feature of the RTE operating system that allows you to access remote files. Also called DS File Transparency.

Transparent Format

One of the file copying formats used by the NS-ARPA/1000 User Service Network File Transfer (NFT). Transparent Format is invoked by default when files are copied between NS-ARPA/1000 systems. It does not alter a file's attributes, but simply copies the file.

transport

A term used to collectively refer to layers 1 through 4 of the OSI model. *See also* **Open Systems Interconnection**.

Transport Layer

Layer 4 of the OSI model. Responsible for end-to-end data integrity. End-to-end indicates that Layer 4 communicates with its peer only at the source and destination nodes, not at intermediate nodes. Layers 5 through 7 also provide end-to-end services, while Layers 1 through 3 are responsible for data integrity between each node. *See also* **Open Systems Interconnection**.

Transport Monitors

NS-ARPA/1000 monitors that process inbound or outbound messages. Transport Monitors act as an interface between the user services and the communication device drivers.

TRC3K

The NS-ARPA/1000 module that formats the data recorded by LOG3K. *See also* **LOG3K**.

TRFAS

See **Transparent File Access**.

TST

See **Transaction Status Table**.

UPLIN

An NS-ARPA/1000 module that cleans up NetIPC resources such as TCBs and user records. It is also the timeout and re-enable module that maintains a running time on all transactions, artificially terminates ("times out") any transaction that is not serviced within a user-specifiable time limit. It can restart any HP-supplied slave monitor that has been aborted, and logoff HP 3000 or HP 1000 sessions whose creating program has terminated with a session still outstanding. *See also* **Transaction Control Block** and **user record**.

user record

See **NetIPC user record**.

Utility Subroutines

A set of NS-ARPA/1000 programmatic calls that enable you to perform special tasks such as downloading absolute or memory-image program files to memory-based nodes and programmatic remote logons to RTE-6/VM and RTE-IV systems.

VCPMN

The NS-ARPA/1000 Virtual Control Panel Monitor. This module monitors the Virtual Control Panel of a remote A/L-Series CPU. VCPMN intercepts and displays VCP messages on the system console of the neighbor node that is the master of the remote system, which is the slave. VCP is a DS/1000-IV Compatible Service.

virtual circuit

See **virtual circuit connection**.

virtual circuit connection

A connection between two NetIPC processes. Virtual circuit connections are the basis for interprocess communication. Once a virtual circuit is established, the NetIPC processes that share it may use it to exchange data. A virtual circuit connection has two major properties: it is a dedicated link, accessible only to the two processes that established the connection; and it provides reliable service, guaranteeing that data will not be corrupted, lost, duplicated, or received out of order.

virtual circuit socket

Used by NetIPC processes to create a virtual circuit connection. Virtual circuit (VC) sockets are the endpoints of a virtual circuit connection.

virtual circuit socket descriptor

Refers to a virtual circuit (VC) socket. A VC socket is the endpoint of a virtual circuit connection between two processes. A VC socket descriptor is returned by the NetIPC calls `IPCRecvCn` and `IPCConnect` after an initial dialogue takes place over a connection formed by call sockets. A NetIPC process can also obtain a VC socket descriptor given away by another process by invoking the NetIPC call `IPCGet`.

watch dogs

NS-ARPA/1000 modules that keep track of internal timing and clean up system resources.

write select

A write select can be performed by using the NetIPC call `IPCSelect.` NetIPC processes can determine whether certain VC sockets are writeable by performing a write select. A writeable socket is one that can immediately satisfy a send request for a number of bytes less than or equal to its write threshold.

write threshold

Used by NetIPC processes that exchange data in asynchronous mode. A NetIPC process determines whether a VC socket is writeable by examining the socket's write threshold. A VC socket is considered writeable if it can immediately satisfy a send request for a number of bytes less than or equal to its write threshold.

X.25

A type of communication link used by NS-ARPA/1000 to provide connections to Packet Switching Networks (PSNs), also known as Value Added Networks (VANs). X.25 links are useful for long-distance communication, and can be more economical than leased lines in some applications. X.25 links have no store-and-forward or dynamic rerouting capabilities. X.25 links can have RTR LIs only and can be used for NS-ARPA and DS/1000-IV compatible services.

X.25 protocol

A protocol used at the Subnet or Intranet Layer (Layer 3s of the NS-ARPA/1000 architecture). X.25 defines some layer 3s functions for messages sent over X.25 links. X.25 networks define routing and store-and-forward features within Packet-Switching Networks.

Bibliography

NS and NS-ARPA/1000-Related Manuals:

NS-ARPA/1000 User/Programmer Reference Manual (91790-90020)
NS-ARPA/1000 Generation and Initialization Manual (91790-90030)
NS-ARPA/1000 Maintenance and Principles of Operation Manual (91790-90031)
NS-ARPA/1000 Quick Reference Guide (91790-90040)
NS-ARPA/1000 Error Message and Recovery Manual (91790-90045)
NS Message Formats Reference Manual (5958-8523)
NS Cross-System NFT Reference Manual (5958-8563)
NS-ARPA/1000 DS/1000-IV Compatible Services Manual (91790-90050)
File Server Reference Guide for NS-ARPA/1000 and ARPA/1000 (91790-90054)
BSD IPC Programmer's Manual for NS-ARPA/1000 and ARPA/1000 (91790-90060)

HP 9000 Manuals:

Using Network Services (B1012-90010)
Using ARPA Services (B1014-90006)

HP 3000 MPE V Manuals:

NS3000/V User/Programmer Reference Manual (32344-90001)
NetIPC 3000/V Programmer's Reference Manual (5958-8581)
NS3000/V Network Manager Reference Manual, Volume I (32344-90002)
NS3000/V Network Manager Reference Manual, Volume II (32344-90012)
NS3000/V Error Message and Recovery Manual (32344-90005)

HP 3000 MPE XL Manuals:

NS3000/XL Operations and Maintenance Reference Manual (36922-61005)
NetIPC 3000/XL Programmer's Reference Manual (36920-61005)
NS3000/XL Error Message Reference Manual (36923-61000)

DS and DS-Related Manuals:

DS/1000-IV User's Manual for RTE-A and RTE-6/VM (91750-90012)
DS/1000-IV Network Manager's Manual Generation and Initialization for RTE-A and RTE-6/VM (91750-90013)
DS/1000-IV Theory of Operation and Troubleshooting for RTE-A and RTE-6/VM (91750-90014)
DS/1000-IV Quick Reference Guide for RTE-A and RTE-6/VM (91750-90015)

X.25 Manuals:

DSN/X.25/1000 Reference Manual (91751-90002)
DSN/X.25/1000 Advanced Guide (91751-90003)
NS X.25 3000/V Link Guide (24405-90002)

RTE-A Manuals:

Getting Started with RTE-A (92077-90039)
RTE-A User's Manual (92077-90002)
RTE-A Driver Reference Manual (92011−90011)
RTE-A Print and Spooling Reference Manual (92077-90248)
RTE-A LINK User's Manual (92077-90035)
RTE-A Programmer's Reference Manual (92077-90007)
RTE-A System Generation and Installation Manual (92077-90034)
RTE-A Backup and Disk Formatting Utilities Reference Manual (92077-90249)

LAN/1000 Manuals:

HP 12076A LAN/1000 Link Local Area Network Interface Controller (LANIC) Installation Manual (12076-90001)
HP 12076A LAN/1000 Link Node Manager's Manual (12076-90002)
HP 12079A LAN/1000 Link Direct Driver Access Manual (12079-90001)

Index