Digital Network Architecture (DNA) Support

symbolics

Cambridge, Massachusetts

Symbolics Digital Network Architecture (DNA) Software Package
998030

February 1986

This document corresponds to Release 6.1 and later releases.

Text written and produced on Symbolics 3600-family computers by the Documentation Group of Symbolics, Inc.
Text typography: Century Schoolbook produced on Symbolics 3600-family computers from Bitstream, Inc. outlines; text masters printed on Symbolics LGP2 Laser Graphics Printers.

Printed in the United States of America.

Printing year and number: 88 87 86 9 8 7 6 5 4 3 2 1

Table of Contents

Page

1. Release Notes for DNA Software Package Release 2.0 **1**
1.1 New Logical Pathname Translation for VAX/VMS V4 Hosts in Release 6.1 2

2. Reference Information on DNA **5**
2.1 Overview of DNA Software Package 5
2.2 Hardware and Software Requirements of DNA 6
2.3 DNA Services 6
2.4 DNA Network Addressing 6
2.4.1 Summary of DNA Addressing Requirements 6
2.4.2 Format of DNA Addresses 7
2.4.3 Mapping a Chaos Address Into a DNA Address 8
2.4.4 DNA Requirement for Ethernet Address 9
2.4.5 Setting the MAX ADDRESS Parameter of the VAX/VMS NCP 11
2.5 Using DNA on the Symbolics Computer 11
2.5.1 Desirability of Network Protocols 11
2.5.2 Connecting to a Remote Host Over the Network 12
2.5.3 Hint for Connecting to Remote Hosts 13
2.5.4 NETWORK Key 13
2.6 DNA Security Considerations 14
2.7 DNA Debugging Tools 15
2.7.1 Using Peek to Check DNA Status 15
2.7.2 Recovering From a Network Problem 16
2.8 References to DECnet Protocol Specifications 16
2.9 DNA Implementation Notes 17
2.10 Adding New Applications That Use DNA 18
2.10.1 Writing a New DNA Application for the Symbolics Computer 18
2.10.2 Writing a New DNA Application for the VAX 21

3. DNA Installation **27**
3.1 Overview of DNA Installation 27
3.2 Assigning DNA and Ethernet Addresses 28
3.3 Using a VAX/VMS Host as a SYS Host 28
3.4 Installing DNA on the First Symbolics Computer 29
3.5 Installing Server Programs on VAX/VMS 31

3.6 Editing the VAX/VMS DECnet Database 32
3.7 Updating the Namespace Database for DNA 32
3.8 Resetting and Enabling the Network for DNA 34
3.9 Verifying the DNA Installation 34
3.10 Installing DNA for Entire Site 35
3.10.1 Configuring Each Symbolics Computer for DNA 35
3.10.2 Loading DNA on a Symbolics Computer 35
3.10.3 Saving and Distributing the DNA World 36

Index **37**

1. Release Notes for DNA Software Package Release 2.0

Release 2.0 of the DNA software package provides the following new features and improvements:

- A CTERM user program is provided for the Symbolics computer, replacing the use of the TELNET protocol for LOGIN service. DNA CTERM is much faster than was DNA TELNET.

- Direct access mode of file openings is supported. That is, the DAP file protocol allows for reading (or writing) from one specified point in a file to another specified point. (Without direct access mode, opening a file for input implies reading the entire file.) See the section "Direct Access File Streams" in *Reference Guide to Streams, Files, and I/O.*

- Documentation on DNA (*Symbolics Digital Network Architecture (DNA) Software Package*) is now available in the Document Examiner, for DNA customers.

- Sites that are upgrading from VAX/VMS V3 to VAX/VMS V4 must be aware of ramifications of the differences in VAX/VMS pathnames. Specifically, the Symbolics computer performs a different logical pathname translation for the two versions of VAX/VMS. See the section "New Logical Pathname Translation for VAX/VMS V4 Hosts in Release 6.1", page 2.

Notes on CTERM

DNA CTERM does not behave exactly the same as TELNET. Users of the previous DNA release might notice one difference in remote login. CTERM supports the VAX/VMS input editing characters, whereas TELNET does not. Therefore, when you are connected to a VAX/VMS system using CTERM, you can use the VAX/VMS input editing characters on a line of text before it gets sent to the VAX/VMS. No characters are sent to the VAX/VMS system until you complete the input, for example, by pressing RETURN .

Notes on Direct Access

Using direct access means to send the messages **:set-pointer** and **:read-bytes** to the file stream. DNA direct access mode has certain restrictions:

- To use direct access on a file, it must be a binary file stored on the VAX/VMS in sequential, fixed-length records, or in the undefined record format.

- You cannot use direct access mode on text files because of character translation, and because character files are stored in RMS-32 record format.

The Symbolics DNA implementation writes all binary files to the VAX in the RMS-32 undefined record format, and character files in RMS-32 record format.

1.1 New Logical Pathname Translation for VAX/VMS V4 Hosts in Release 6.1

VAX/VMS V4 implements a new and more flexible pathname scheme than does VAX/VMS V3. In brief, pathnames in V4 can be longer, and can contain the underscore character "_". The underscore character is a convenient way to separate parts of a filename, much as a hyphen is used on other operating systems.

Logical pathnames are translated differently for the two versions of VAX/VMS. For example, when the Symbolics computer translates a logical pathname to a VAX/VMS V3 host, it often must compress the pathname to make it adhere to the V3 restrictions on the length of pathname components. In addition, the Symbolics computer removes any underscore characters because they are illegal in V3 pathnames. However, in translating a logical pathname to a VAX/VMS V4 pathname, the Symbolics computer need not compress the filename (as much; V4 does have some length restrictions) nor remove underscores.

To determine which way to translate a logical pathname, a Symbolics computer first consults the host object for the VAX in the namespace database to determine whether the host is running VAX/VMS V3 (the System Type is VMS) or VAX/VMS V4 (the System Type is VMS4).

This pathname difference can cause problems only if you change the System Type in the host object in the namespace from VMS to VMS4 *after* having installed Symbolics sources and documentation files (while the System Type in the host object was set to VMS). At the point when you change the host object to System Type VMS4, logical pathnames will be translated differently. When the Symbolics computer tries to find a source or documentation file resident on the VAX, it might not be able to find that file.

You have three possible ways of handling the VAX/VMS upgrade:

- Coordinate the VAX/VMS upgrade with a Symbolics software upgrade. Perform the VAX/VMS upgrade first, then change the System Type in the VAX's host object, then install the new Symbolics software and documentation files.

- Immediately after you upgrade to VMS V4, change the System Type in the VAX's host object, and then reinstall all Symbolics source and documentation files that reside on the VAX/VMS system.

- Upgrade to VMS V4, but do not change the System Type in the host object, and do not reinstall the sources and documentation files. Logical pathname translation will continue to work as if VMS filenames still followed the V3 rules. This presents some problems:
    - ° You will be unable to take advantage of the greater flexibility of pathnames in V4, from the Symbolics computer.
    - ° You will be unable to access files on the VAX that do take advantage of the V4 flexibility. That is, any files with long pathname components or underscores will be unavailable to the Symbolics computer.

To reinstall the Symbolics sources and documentation files, refer to the *Software Installation Guide.* See the section "Upgrading a Machine from Release 6.0 to Release 6.1". Follow the step called "Load the files from the documentation tape and the source tape."

You must also reinstall any optional Symbolics software used at your site, such as:

- VAX/VMS Chaosnet Software Package
- Digital Network Architecture (DNA) Software Package
- FORTRAN Software Package
- PASCAL Software Package

2. Reference Information on DNA

This section contains background information on DNA. It does not contain any installation instructions. For step-by-step instructions on installing the DNA system: See the section "DNA Installation", page 27.

2.1 Overview of DNA Software Package

The Digital Network Architecture (DNA) software package enables the Symbolics computer to access services provided by a VAX/VMS systems using the DNA protocols. These systems can be located either on the local Ethernet or on some other DNA network connected to the local Ethernet via a router node. The DNA application protocols supported include:

- Electronic mail (DNA-MAIL)
- Remote login (CTERM)
- File transfer (DAP)

Symbolics does not support the use of DNA protocols between two Symbolics computers.

The primary goal of the Symbolics DNA support package is: to enable the VAX/VMS machine to provide services (such as FILE, LOGIN, and MAIL services) to the Symbolics computer using DECnet protocols. Symbolics computers support DNA user programs that communicate with DNA server programs on the VAX/VMS machine.

Note that the DNA package does not enable the Symbolics computer to provide all of the same services to VAX/VMS users. Symbolics computers offer the following services to the VAX/VMS users:

- Electronic mail (DNA-MAIL)
- File transfer (DAP)
- A LOOPBACK server, used typically in testing the communication path

Remote login using CTERM is supported in one direction only. You can log in to the VAX/VMS machine from a Symbolics computer using DNA CTERM; however, you cannot log in to a Symbolics computer from VAX/VMS using DNA CTERM.

2.2 Hardware and Software Requirements of DNA

DNA Release 2.0 runs on Symbolics 3600-family Lisp Machines running Release 6.1 software.

The DNA package requires a VAX running VAX/VMS version 4.0 or higher, with VAX/VMS DECnet Phase IV or higher. The VAX must be connected to the Ethernet, or have access to a VAX router node on the Ethernet.

2.3 DNA Services

The DNA software package enables Symbolics computer users to access the following services provided by the VAX:

Service	*Medium*	*Protocol*
MAIL-TO-USER	DNA	DNA-MAIL
FILE	DNA	DAP
TIME	DNA	DNA-LMTIME
LOGIN	DNA	CTERM
UPTIME	DNA	DNA-LMUPTIME
TAPE	DNA	RTAPE
SHOW-USERS	DNA	ASCII-NAME

Note that the DNA package does not enable the Symbolics computer to provide the same services to VAX/VMS users. The Symbolics computer offers the following services to the VAX/VMS users:

Service	*Medium*	*Protocol*
FILE	DNA	DAP
MAIL-TO-USER	DNA	DNA-MAIL
LOOPBACK	DNA	DNA-LOOPBACK-MIRROR

2.4 DNA Network Addressing

2.4.1 Summary of DNA Addressing Requirements

The following requirements must be met, for hosts at a site to be able to communicate using DNA:

- Each host that will use DNA protocols requires a valid and unique DNA address, whether the machine is a VAX or a Symbolics computer.

- The DNA address of each VAX and Symbolics computer must be stored in its host object in the Namespace database.

- Hosts can communicate with one another only if they have the same area number.

- Each host must have a valid Ethernet address, which must be derived from the DNA address.

- Symbolics computers need to have their Ethernet addresses set in their boot files.

- The VAX NCP contains a parameter called MAX ADDRESS. This is the highest DNA node number that the NCP will recognize. It is important that no DNA address exceeds this limit. By default, MAX ADDRESS is set to 32. If you have node number higher than 32, it is a simple matter to set the MAX ADDRESS parameter to a higher value.

This document contains some suggestions for handling DNA addressing. These suggestions are by no means requirements. Each site can judge whether these suggestions make it more convenient to administer DNA, and use them or not. The suggestions are:

- You can assign DNA addresses to hosts based on their Chaos addresses. This mapping scheme has two advantages: it ensures that the DNA address is unique; it saves you from maintaining two distinct and unrelated lists of addresses (one for DNA, one for Chaos).

- You can enter the DNA addresses of the Symbolics computers to the VAX/VMS DECnet database. This is not strictly necessary. If you do so, the VAX/VMS is kept informed of all the DNA hosts on the network. Then if a VAX/VMS user requests a service from a Symbolics computer, the request can be fulfilled. Typically, the Symbolics computer requests a service from the VAX/VMS host; this does not require that the VAX/VMS DECnet database be current.

2.4.2 Format of DNA Addresses

DNA addresses have two components: an area and a node number in that area. For example, a DNA address of 3.7 indicates the host is node 7 in area 3. Hosts with different area numbers are prevented from communicating with each other.

DNA addresses are 16 bit quantities, where the high-order 6 bits constitute the area, and the low-order 10 bits constitute the node number. DNA addresses are expressed in decimal notation.

Example: DNA Address 3.7

```
 15  14  13  12  11  10  9   8   7   6   5   4   3   2   1   0
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

|<---------Area-------->|<------------Nodenumber------------->|
```

Bits 0-9 represent the node number, in this example 7.
Bits 10-15 represent the area number, in this example 3.

You can choose DNA addresses for your hosts in any way you like, as long as:

- Each host that will use DNA protocols, whether the machine is a VAX or a Symbolics computer, must have a valid and unique DNA address.
- Any two hosts that want to communicate with each other are in the same area. For example, the Symbolics computer area numbers must be the same as the area number for any VAX that is a server machine.

Some sites choose to assign DNA addresses sequentially, from 1.1, 1.2, 1.3 and so on.

You can also choose the DNA addresses based on the Chaos addresses: See the section "Mapping a Chaos Address Into a DNA Address", page 8.

2.4.3 Mapping a Chaos Address Into a DNA Address

We recommend that you choose DNA addresses for the hosts at your site based on the Chaos addresses. Each Symbolics computer already has a unique Chaos address. By choosing a DNA address derived from the Chaos address, you can always determine a DNA address from the Chaos address (thus assuring that the DNA address is unique), and you can derive the Chaos address from the DNA address. This saves you from maintaining two distinct and unrelated addressing schemes, one for Chaos and one for DNA.

It is not necessary or required that you derive DNA addresses based on the Chaos addresses. This is just a suggestion.

Some sites cannot use this mapping scheme. If your site has several VAX/VMS hosts that are already using DNA protocols, they already have DNA addresses assigned to them. In that case, you must be sure to assign DNA addresses to the Symbolics computers that have the same DNA area number as the VAX/VMS hosts on the network. These addresses must be unique within the DNA database.

If you use this mapping scheme, keep in mind that the node numbers of each host must be below the VAX's limit, which is the MAX ADDRESS parameter of the

NCP. The NCP does not accept network communication from hosts with node numbers higher than MAX ADDRESS. By default, MAX ADDRESS is 32. It is an easy matter to set the MAX ADDRESS higher.

Start by figuring out the Chaos address of the first host to have DNA installed on it. You can do this by entering the namespace database (choose it from the System menu): use [View], then use [Host], then enter the name of the host. Each Symbolics computer host object should contain a Chaos address (expressed in octal notation) that resembles:

```
Address: Pair: CHAOS 401
```

To map a Chaos address into a DNA address, first determine the Chaos host number and subnet number from the address. The Chaos host number is be the DNA node number. The Chaos subnet number is the DNA area number.

Chaos Address 401

```
 15  14  13  12  11  10  9   8   7   6   5   4   3   2   1   0
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

|<----Chaos Subnet Number------>|<-----ChaosHost Number------->|
```

The Chaos subnet number is 1.
The Chaos host number is 1.
The Chaos address is 401 octal.

In this example, the Chaos subnet number is 1, so the DNA area number is 1. The Chaos host number is 1, so the DNA node number is 1. The Chaos address 401 maps into a DNA address of 1.1.

Note that this mapping of Chaos subnet number to DNA area number works only if the Chaos subnet number uses six or less of the available eight bits, that is, if the Chaos subnet number is 128 or less. Any Chaos address that is 37777 or less can be fully mapped into a DNA address. Chaos addresses greater than 37777 can be partially mapped into DNA addresses, by mapping only the Chaos host number into the DNA node number.

2.4.4 DNA Requirement for Ethernet Address

To use the DNA network protocols, each Symbolics computer must have a 48-bit Ethernet address set in its boot file. This is a requirement of the design of DNA. The Ethernet address is derived from the DNA address already assigned to the host.

The DNA installation instructions contain a function you can type in to a Lisp Listener and use to determine the Ethernet address for a given DNA address. See the section "Assigning DNA and Ethernet Addresses", page 28.

This section describes the format of an Ethernet address, and the method of mapping a DNA address to an Ethernet address.

Each Symbolics computer using DNA must have the following line in the boot file on the FEP file system. xx and yy are placeholders, that are derived from the host's DNA address:

```
Set Ethernet-address AA-00-04-00-xx-yy
```

If you are assigning DNA addresses sequentially from 1.1 on, and your first Symbolics computer has DNA address 1.1, the corresponding DNA address is AA-00-04-00-01-04.

DNA addresses are 16 bits, where the high-order 6 bits constitute the area, and the low-order 10 bits constitute the node number. The Ethernet address is in the format: AA-00-04-00-xx-yy. The "AA-00-04-00" part of the address is the literal value in all cases. The xx and yy fields of the Ethernet address are placeholders whose values are derived from the DNA address; they are the values of the DNA address, byte-swapped, represented in hexadecimal.

Example: DNA Address 3.7

```
 15  14  13  12  11  10  9   8   7   6   5   4   3   2   1   0
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

|<---------Area-------->|<------------Nodenumber-------------->|

|<--------Value of yy---------->|<----------Valueof xx-------->|
```

Bits 0-9 represent the node number, in this example 7.
Bits 10-15 represent the area number, in this example 3.

Bits 0-7 are the value of xx, in this example 07 (hexadecimal).
Bits 8-15 are the value of yy, in this example 0C (hexadecimal).

Therefore, a host with DNA address 3.7 has this line in the boot file:

```
Set Ethernet-address AA-00-04-00-07-0C
```

Once you have added the command to the boot file that sets the host's Ethernet address, you must cold boot the machine in order to make the Ethernet address take effect.

If you are familiar with other protocols, such as Chaos and IP/TCP, you might be interested in knowing some background information on the DNA Ethernet address requirement. Unlike DNA, neither Chaos nor IP/TCP requires the Symbolics computer to set an Ethernet address in its boot file. Instead, Chaos and IP/TCP use a dynamic method of determining a host's Ethernet address, called the Address Resolution Protocol. Chaos and IP/TCP query hosts for their Ethernet address, and receive a reply. All of this happens invisibly to the user. DNA Phase IV made no provision for use of the ARP, and therefore requires a static Ethernet address for each host. This might change in a future phase of the architecture.

2.4.5 Setting the MAX ADDRESS Parameter of the VAX/VMS NCP

The VAX/VMS NCP program controls the network communication on the VAX side. The NCP contains a parameter called MAX ADDRESS, which is the highest node number that it will recognize. The NCP does not accept network communication from hosts with node numbers higher than MAX ADDRESS. By default, MAX ADDRESS is 32. It is an easy matter to set MAX ADDRESS higher, if you have node numbers that are higher.

You can enter the NCP program and type:

```
DEFINE EXEC MAX ADDRESS 255
```

This sets MAX ADDRESS to 255 in the permanent database. Note that if you have node numbers higher than 255, you can set MAX ADDRESS to a higher value.

2.5 Using DNA on the Symbolics Computer

Once the DNA software package is installed and DNA is loaded, DNA protocols are used automatically by the Symbolics computer whenever they are appropriate for performing some network service. For example, DNA/CTERM is used by the Terminal program, DNA/DAP is used when remote files are opened, and DNA/ASCII-NAME is used by the Show Users command.

You might notice one difference in using DNA compared to using Chaos or IP/TCP. The DNA protocol itself requires you to identify yourself with a valid password. For more information: See the section "DNA Security Considerations", page 14.

2.5.1 Desirability of Network Protocols

When you request a network service, such as FILE service, the Symbolics generic network system chooses a *path* to that service. When more than one path to the service exists, the generic network system tries to choose the most efficient path. One of the factors taken into account is the *desirability* of the various protocols that support the service. The implementor of the protocol assigns a number between 0 and 1 to be the desirability of the protocol.

The relative desirability factors of the various Symbolics network protocols are as follows:

- Chaos protocols have the highest desirability.
- IP/TCP protocols are less desirable than Chaos.
- DNA protocols are less desirable than IP/TCP.

The one exception to the above rules is the NFILE protocol. NFILE used over the TCP medium has a higher desirability than does NFILE over CHAOS.

Although the desirability factor of the protocols is built into the implementation, you can alter the desirability factors at your site.

The site object for your site in the namespace database has an attribute called **host-protocol-desirability**. By entering values for that site attribute, you can specify a tuning factor that overrides the Symbolics default desirability for a protocol. For further information: See the section "**host-protocol-desirability**: Site Object Attribute" in *Networks*.

For other information related to the desirability of network protocols:

- See the section "Service Descriptions: Lisp Machine Generic Network System" in *Networks*.
- See the section "Interfacing to the Service Lookup Mechanism" in *Networks*.
- See the section "Users: Defining Protocols: Lisp Machine Generic Network System" in *Networks*.

2.5.2 Connecting to a Remote Host Over the Network

If your Symbolics computer is on a network and configured properly, you can access other hosts on the network with the Terminal program.

To use the Terminal program, press SELECT T. The prompt is:

```
Connect to host:
```

Type the name of the host to which you want to connect. The network system consults the namespace database to determine the best route to the host, provided that it supports LOGIN service. (You can also press HELP in response to the initial prompt, to read input editor documentation.)

Once you are connected to the remote host, that host displays its prompt on the screen. You are now communicating directly with the remote machine.

When you are connected to a foreign host, the NETWORK key provides several useful commands. You can press NETWORK and then HELP for a list of these commands. See the section "NETWORK Key", page 13. For example, you can log out of the remote host and break the connection by pressing NETWORK L.

If you want to use the Terminal program to log in to a remote Symbolics computer when someone is logged in to that machine, you must first enable remote login by evaluating the form **(net:remote-login-on)** on that machine. See the function **net:remote-login-on** in *Networks*.

2.5.3 Hint for Connecting to Remote Hosts

If you frequently connect to remote hosts, you might benefit by including the following form in your init file:

```
(setq telnet:*nvt-overstrike-default*nil)
```

This form changes the default network Overstrike mode to be No. By including the form in your init file, the default for Overstrike is set to No during login.

When Overstrike is set to No, characters are erased from the screen when you delete them from a command line (when connected to a remote host). When Overstrike mode is set to Yes, the deleted characters remain on the screen and any new characters you type are written directly over them. This makes it difficult to read the characters.

You can also toggle Overstrike mode by using the NETWORK X command, and clicking on the appropriate menu item.

2.5.4 NETWORK Key

This key is used to get the attention of the Terminal program. You must be connected to a host via the Terminal program before you can use this key. See the section "Connecting to a Remote Host Over the Network", page 12.

Once connected, commands are given by pressing NETWORK and another single character.

The following commands are available:

NETWORK HELP	Display the list of options for the NETWORK key.
NETWORK A	Send an ATTN (in Telnet, a new Telnet "Interrupt Process").
NETWORK D	Disconnect without logging out first.
NETWORK L	Log out of remote host, and break the connection.
NETWORK Q	Quit, by disconnecting and deselecting this window.
NETWORK M	Toggle More processing.

`NETWORK X` Enter an extended network command; see below.

More complicated commands are entered with the extended command, `NETWORK X`. This command invokes a `Choose Variable Values` window.

`NETWORK X` provides the capability to tailor the following:

- The escape character. The default is `NETWORK` .
- Whether characters overstrike or erase. Characters erase by default.
- Whether More processing is enabled. More processing is enabled by default.
- Whether to enable the *wallpaper* facility, which logs host output to a file. By default, wallpaper is not enabled.
- The filename of the wallpaper file.
- For Telnet, what level of filtering and interpretation is placed on the characters; for example, whether Imlac terminal codes are interpreted in host output.

2.6 DNA Security Considerations

DNA requires access control information on a user on most requests for connection, regardless of whether the application program you are requesting requires a password. The only exception to that rule is for LOGIN service.

Therefore, the first time you connect to a foreign host, DNA itself requires a valid password for that host. This occurs when you use any of the DNA services (except LOGIN), even simple ones such as TIME and UPTIME.

By default, DNA keeps a password cache on the local Symbolics Lisp Machine. That is, each time you enter a DNA password for a certain host, a new entry is added to the password cache containing your user name, the host name, and the password. In future connections, DNA looks up your user name and the host name in its cache to see if there is a password to be used. Thus, you are not prompted for a password when you next request a connection to a certain host.

dna:enable-password-caching *Function*

DNA password caching is enabled by using this function. When it is enabled, for each DNA connection to a foreign host, DNA looks for the current user and host in its password cache (stored locally on the Symbolics computer). If this user has logged into the same host, DNA can enter the password found in the cache. If the user and host do not appear in the password cache, DNA prompts for a password, uses it in this connection, and stores it in the cache.

DNA has password caching enabled by default.

See the function **dna:flush-password-cache**, page 15.
See the function **dna:disable-password-caching**, page 15.

dna:disable-password-caching *Function*

DNA password caching is disabled by using this function. When it is disabled, the DNA password cache is flushed, using **dna:flush-password-cache**. DNA does not cache any DNA passwords, so the user is prompted for a password on each request for connection.

See the function **dna:flush-password-cache**, page 15.
See the function **dna:enable-password-caching**, page 14.

dna:flush-password-cache *Function*

This function destroys the DNA password cache stored on the local Symbolics computer. If password caching was in effect previously, it is still in effect. Flushing the cache has no effect on whether DNA password caching is enabled or disabled.

See the function **dna:disable-password-caching**, page 15.
See the function **dna:enable-password-caching**, page 14.

2.7 DNA Debugging Tools

2.7.1 Using Peek to Check DNA Status

Once DNA is installed and loaded, you can check its status with the Peek facility. For a description of Peek: See the section "Using Peek" in *User's Guide to Symbolics Computers.*

Peek is available from the System menu, or by using SELECT P Click the mouse on the Network heading. By clicking on Meters, under DNA, you see the values of various meters collected by the Symbolics computer. The display is continually updated. You can make a simple check of the status of DNA by looking at the values of *ENDNODE-HELLOS* and *ROUTER-HELLOS* in the Peek display.

Each DNA host on the network periodically sends *hello messages* to other DNA hosts on the network. Each DNA host receives hello messages from other hosts on the network; the total number of hellos received is the sum of *ROUTER-HELLOS* and *ENDNODE-HELLOS*. In a normal situation, the number of hellos received (*ROUTER-HELLOS* + *ENDNODE-HELLOS*) should be positive. If the two numbers are zero, the local host is not communicating at all with other DNA hosts.

2.7.2 Recovering From a Network Problem

The symptom of a network problem is the inability of your local Symbolics computer to communicate with another host (or other hosts) on the network. For example, you cannot transfer a file from a remote host to your local machine. Try one of the following suggestions:

2.7.2.1 Resetting and Enabling the Network

Reset Network

Turns your network interface off and back on again. This is useful if your connections appear to be stuck and nothing is being transmitted or received.

2.7.2.2 Resetting File Access Paths

If you are having a problem with file transfer to a certain host, you can reset the file access path to that host instead of using the Reset Network command, which would reset all file access paths.

Choose the Peek program by pressing SELECT P. Click on the heading File Systems. The resulting display has an entry for each foreign host to which this host has an open file access path.

To reset a specific file access path, position the mouse over the name of the host that is experiencing the problem. Click right to get a menu, then use [File Reset].

Alternatively, you can type the following form to a Lisp Listener, substituting the name of the problem host for "*YOUR-HOST*".

```
(send (si:parse-host "YOUR-HOST") :file-reset 't))
```

2.8 References to DECnet Protocol Specifications

These documents are available from Digital Equipment Corporation:

Software Documentation
1925 Andover Street TW/E07
Tewksbury, Massachusetts 01876

- *DECnet Digital Network Architecture (Phase IV) General Description*, Order No. AA-N149A-TC

- *DECnet Digital Network Architecture (Phase IV) Ethernet Node Product Architecture Specification*, Order No. AA-X440A-TK

- *DNA Session Control Functional Specification*, Version 1.0.0, Order No. AA-K182A-TK

- *DNA Data Access Protocol (DAP) Functional Specification*, Version 5.6.0, Order No. AA-K177A-TK
- *DNA Routing Layer Functional Specification*, Version 2.0.0, Order No. AA-X435A-TK
- *DNA Network Services Protocol (NSP) Functional Specification*, Version 4.0.0, Order No. AA-X439A-TK
- *Guide to Networking on VAX/VMS*, Order No. AA-Y512A-TE

2.9 DNA Implementation Notes

Symbolics DNA is an *endnode* implementation of DNA Phase IV, that also conforms to the generic Symbolics generic network model. An endnode is a host that sends messages directly to its adjacent nodes on the Ethernet. An endnode communicates only over the Ethernet to which it is attached, using other DNA nodes on this same Ethernet as designated routers to communicate with nodes that are not connected to the Ethernet.

DNA is a layered architecture that maps to the International Standards Organization reference model for Open Systems Interconnection (ISO/OSI). The Symbolics DNA Package implements the following DNA layers, all Phase IV:

- DNA Routing layer - corresponding to layer 3 (*network*) of ISO/OSI
- DNA Network Services layer - layer 4 (*transport*) of ISO/OSI
- DNA Session Control layer - layer 5 (*session*) of ISO/OSI
- Several application layer protocols - layers 6 (*presentation*) and 7 (*application*) of ISO/OSI

The Data Link (layer 2 of ISO/OSI) and the Physical Link (layer 1 of ISO/OSI) are already implemented by the Ethernet interface support modules in the Symbolics computer.

See the section "References to DECnet Protocol Specifications", page 16.

2.10 Adding New Applications That Use DNA

This section is intended for programmers interested in writing new applications built on the DNA software package.

You can write programs to extend the DNA capability. In other words, you can add a new service that uses the DNA medium. In the terminology used by Digital Equipment Corporation, this is *task-to-task communication*, meaning some sort of communication between two arbitrary application programs over a network link.

The first example is a COMMAND server that enables a Symbolics computer user to give a command to be executed on the VAX. The second example is an EVAL server that enables a VAX user to submit a Lisp form to be evaluated on a Symbolics computer.

Please note that these examples are intended only to illustrate how to write applications using DNA. As presented in this section, these are not robust programs.

2.10.1 Writing a New DNA Application for the Symbolics Computer

This section illustrates adding a simple application that uses a DNA byte-stream to allow a user on a Symbolics computer to invoke a new service on the VAX. The new service is COMMAND service; it invokes a single VMS DCL command on the VAX.

To implement COMMAND service, we need to write a user side that runs on a Symbolics computer, and a server side that runs on the VAX.

The User Side of COMMAND

The user side consists of a protocol definition (using **neti:define-protocol**) and a call to the function **dna:add-dna-contact-id-for-protocol**, to inform the DNA system what to use as a contact ID when sending its connection initiation request to the VAX. Here is the Lisp code:

```
;;; Define the protocol - its generic service is :COMMAND
```

```
(NETI:DEFINE-PROTOCOL :DNA-COMMAND-EXECUTE (:COMMAND :BYTE-STREAM)
  (:DESIRABILITY .9)
  (:INVOKE (PATH)
    (WITH-OPEN-STREAM (STREAM (NETI:GET-CONNECTION-FOR-SERVICEPATH))
      (LET* ((ARGS (NETI:SERVICE-ACCESS-PATH-ARGSPATH))
             (COMMAND (CAR ARGS)))
        (UNLESS (STRINGP COMMAND)
          (FERROR "First argument should be a string."))
        (SEND STREAM :LINE-OUT COMMAND)
        (SEND STREAM :FORCE-OUTPUT)
        ;; Now wait for response from the server - this should
        ;; be just an echo of the command that we sent.
        (LET ((RESPONSE (SEND STREAM :LINE-IN)))
          (IF (STRING-EQUAL RESPONSE COMMAND)
              "Command Received Successfully"
              "Command Not Received Successfully"))))))

;;; Add the contact name for the DNA system to use in its request
;;; for a network connection on the VAX.

(DNA:ADD-DNA-CONTACT-ID-FOR-PROTOCOL' :DNA-COMMAND-EXECUTE "COMMAND")
```

Note that you can undo the effect of the previous form as follows:

```
(DNA:REMOVE-DNA-CONTACT-ID-FOR-PROTOCOL' :COMMAND)
```

We need to add COMMAND service to the host object for the VAX in the namespace database. The resulting entry looks like:

`Service:` **Set:** `COMMAND DNA DNA-COMMAND-EXECUTE` *Global-name*

The Server Side of Command

The code below is the VMS Macro-32 source (called COMMAND.MAR) for the server program that runs on the VAX side:

```
;*************Beginningof COMMAND server program********

.title decnet COMMAND server

.macro errcheck,?lbl
       blbs    r0,lbl
       $exit_s r0
lbl:
.endm errcheck

lognam:: .ascid /SYS$NET/
netchan:: .blkw 1
netiosb:: .blkl 2
```

```
command_desc::
        .word   132,0
        .address command_string
command_string::
        .blkb   132

.entry start,0

;;; Get the network channel
        $assign_s devnam=lognam,-
                  chan=netchan
        errcheck

;;; Read the command from the network
        $qiow_s chan=netchan,-
                func=#io$_readvblk,-
                iosb=netiosb,-
                p1=command_string,-
                p2=command_desc
        errcheck

;;; Setup the command string and descriptor
        movw    netiosb+2,command_desc          ;Length

;;; Acknowledge receipt of the command
        $qiow_s chan=netchan,-
                func=#io$_writevblk,-
                iosb=netiosb,-
                p1=command_string,-
                p2=command_desc
        errcheck

;;; Deassign the network channel
        $dassgn_s chan=netchan

;;; Execute the command
        decw    command_desc
        pushal  command_desc
        calls   #1,g^lib$do_command
        $exit_s r0

.end start
;*************End of COMMAND server program********
```

We assemble and link this file with following VMS DCL commands:

```
$ MACRO COMMAND.MAR
$ LINK COMMAND.OBJ
```

We then copy the resulting image to SYS$SYSTEM:COMMAND.EXE with the following VMS DCL command:

```
$ COPY COMMAND.EXE SYS$SYSTEM:COMMAND.EXE
```

We now create a file called COMMAND.COM in the SYS$SYSTEM directory that contains only the following VMS DCL command:

```
$ RUN SYS$SYSTEM:COMMAND.EXE
```

And finally, we inform VMS DECnet of the existence of this server by typing the following VMS DCL command:

```
$ MCR NCP SET OBJECT COMMAND NUMBER 0 FILE SYS$SYSTEM:COMMAND.COM
```

The above command requires certain VAX/VMS privileges.

Using the New COMMAND Service

To invoke this command server from a Symbolics computer, we use the function **neti:invoke-service-on-host** with the appropriate arguments, as in this example:

```
(NETI:INVOKE-SERVICE-ON-HOST':COMMAND (SI:PARSE-HOST "Dumbo")
                                      "$ RUN FOO.EXE")
```

Since the command is actually executed after the network link has been closed, we can verify that the execution was successful by checking the NETSERVER.LOG file on the VAX.

2.10.2 Writing a New DNA Application for the VAX

This section illustrates adding a simple application that uses a DNA byte-stream to allow a user on a VAX computer to invoke a new service on the Symbolics computer. The new service is EVAL service; it invokes the Symbolics computer **eval** function.

To implement EVAL service, we need to write a user side that runs on a VAX, and a server side that runs on the Symbolics computer.

Note that the EVAL server runs on the Symbolics computer only if no one is currently logged into the machine. Keep in mind also that the Symbolics DECnet implementation does not provide any kind of access control or security features.

The Server Side of EVAL

The following code is the EVAL server that resides on the Symbolics computer:

```
;;;; EVAL server
;;; Values can be T, :NOTIFY or NIL
(DEFVAR EVAL-SERVER-ON NIL)

(DEFUN EVAL-SERVER-ON (&OPTIONAL (MODE T)) (SETQ EVAL-SERVER-ON MODE))
```

```
(DEFUN EVAL-SERVER-FUNCTION (PROTOCOL SERVER-ON STREAM SERVER)
  (*CATCH 'EVAL-SERVER-EXIT
    (COND ((AND (NULL SERVER-ON) (NOT (EQUAL USER-ID "")))
           (SEND STREAM ':REJECT (FORMAT NIL "This machine is in use by ~A" USER-ID))
           (*THROW 'EVAL-SERVER-EXIT NIL))
          ((EQ SERVER-ON ':NOTIFY)
           (TV:NOTIFY NIL "Use of ~A server by ~A"
                      PROTOCOL (HOST-SHORT-NAME (SEND STREAM ':FOREIGN-HOST)))
           (PROCESS-ALLOW-SCHEDULE)
           (SEND STREAM ':ACCEPT))
          (T
           (SEND STREAM ':ACCEPT)))
    ;; Rescue machine in case of lossage.
    (CATCH-ERROR
      (WITH-OPEN-STREAM (STREAM STREAM)
        (FUNCALL SERVER STREAM))
      NIL)))

;;; Note that the connection will be closed if there is a read-time error
;;; since the call to READ lies outside the CATCH-ERROR.
(NET:DEFINE-SERVER :EVAL (:MEDIUM :BYTE-STREAM :REJECT-UNLESS-TRUSTED T :WHO-LINE T
                      :STREAM (STREAM :ACCEPT-P NIL))
  (EVAL-SERVER-FUNCTION ':EVAL EVAL-SERVER-ON STREAM #'EVAL-SERVER))

(DEFUN EVAL-SERVER (TERMINAL-IO)
  (DO ((INPUT)) (NIL)
    (IF (EQ (SETQ INPUT (READ TERMINAL-IO 'QUIT)) 'QUIT)
        (*THROW 'EVAL-SERVER-EXIT NIL))
    (CONDITION-CASE (ERROR)
        (PRINT (MULTIPLE-VALUE-LIST (EVAL INPUT)))
      ((ERROR PDL-OVERFLOW)
       (TERPRI)
       (PRINC ERROR)))
    (SEND TERMINAL-IO ':TYO #\NEWLINE)
    (SEND TERMINAL-IO ':FORCE-OUTPUT)))
```

Now, the Symbolics implementation of the DNA network must be told what server corresponds to the contact id that the VMS program will use to request the EVAL service:

```
(DNA:ADD-DNA-CONTACT-ID-FOR-PROTOCOL ':EVAL "EVAL")
```

Note that you can undo the effect of the previous form as follows:

```
(DNA:REMOVE-DNA-CONTACT-ID-FOR-PROTOCOL ':EVAL)
```

The User Side of EVAL

The code below is the VMS Macro-32 source (called EVAL.MAR) for the user side of this application, that runs on the VAX:

```
;*************Beginningof EVAL user program********

.Title VMS User side for Symbolics Computer EVAL server

.macro errcheck,?lbl
        blbs    r0,lbl
        $exit_s r0
lbl:
.endm errcheck

target_desc::
        .word   32,0
        .address target_string
target_string::
        .blkb   32
task_string::
        .ascii  /::"TASK=EVAL"/
task_length == . - task_string

node_prompt::   .ascid /Host: /
node_length::   .long  0

form_prompt::   .ascid /Form: /
form_length::   .long  0

netchan:: .blkw 1
netiosb:: .blkl 2

form_desc::
        .word   512,0
        .address form_string
form_string::
        .blkb   512

response_desc::
        .word   512,0
        .address response_string
response_string::
        .blkb   512
```

```
.entry start,0
;;; Get the remote node
        pushal  node_length
        pushal  node_prompt
        pushal  target_desc
        calls   #3,g^lib$get_input
        errcheck

;;; Copy the remote node into the target string
        moval   task_string,r6
        movzwl  node_length,r7
        subl    r7,r6
        movc3   r7,target_string,(r6)
        movl    r6,target_desc+4
        addw3   node_length,#task_length,target_desc

;;; Get the network channel - request a connection from remote node
        $assign_s devnam=target_desc,-
                  chan=netchan
        errcheck

;;; Get the form to be EVAL'ed from the command line, or prompt user
        pushal  form_length
        pushal  form_prompt
        pushal  form_desc
        calls   #3,g^lib$get_input
        errcheck

        movw    form_length,form_desc

;;; Send the form over to the remote node
        $qiow_s chan=netchan,-
                func=#io$_writevblk,-
                iosb=netiosb,-
                p1=form_string,-
                p2=form_desc
        errcheck


;;; Read the response from the remote node
        $qiow_s chan=netchan,-
                func=#io$_readvblk,-
                iosb=netiosb,-
                p1=response_string,-
                p2=response_desc
        errcheck
```

```
;;; Print out the result for the user
        pushal  response_desc
        calls   #1,g^lib$put_output
        errcheck

;;; Deassign the network channel
        $dassgn_s chan=netchan
        $exit_s r0

.end start
;*************End of EVAL user program********
```

We assemble and link this file with following VMS DCL commands:

```
$ MACRO EVAL.MAR
$ LINK EVAL.OBJ
```

Using the New EVAL Service

First, ensure that no one is logged in to the Symbolics computer you will use as the EVAL server.

From the VAX, give the command:

```
$ RUN EVAL.EXE
```

The program prompts you for the name of the server host and the Lisp form to be evaluated.

3. DNA Installation

This section contains instructions for installing the DNA system and configuring the site for DNA. For further information on DNA: See the section "Reference Information on DNA", page 5.

Sites that are planning on upgrading from VAX/VMS V3 to V4 should be aware of ramifications of the differences in VAX/VMS pathnames. Specifically, the Symbolics computer performs a different logical pathname translation for the two versions of VAX/VMS. This difference can result in problems if files are installed on the VAX using the V3 logical translation rules and later the site begins to use the V4 rules. The Symbolics computer might be unable to locate some of the files stored on the VAX. For more information: See the section "New Logical Pathname Translation for VAX/VMS V4 Hosts in Release 6.1", page 2.

3.1 Overview of DNA Installation

The following sections describe the process of installing DNA at your site. You must have completely installed Release 6.1 before beginning the installation of DNA. This installation involves loading a tape on the VAX/VMS system and loading a cartridge tape on a Symbolics computer.

Sites intending to use the VAX/VMS host as a SYS host need to perform some extra installation steps. See the section "Using a VAX/VMS Host as a SYS Host", page 28.

The order of the steps in the DNA installation must be done as described in this section. If certain steps are done out of order, the installation will not be successful.

The installation contains the following steps:

1. Assigning DNA and Ethernet addresses to hosts.

2. Installing DNA on one Symbolics computer.

    - Editing boot file to set its Ethernet address.
    - Restoring DNA from the distribution tape.
    - Creating a network object for DNA in the namespace database.

3. Installing server programs on the VAX/VMS host.

4. Editing the VAX/VMS DECnet database.

5. Updating the namespace database for DNA.

6. Resetting and enabling the network for DNA.

7. Verifying that the installation was successful.

8. Installing DNA on other Symbolics computers at your site.

3.2 Assigning DNA and Ethernet Addresses

You must assign a valid and unique DNA address to each host that will be using the DNA protocols. For details on topics related to DNA addressing: See the section "DNA Network Addressing", page 6.

Because DNA protocols require a static Ethernet address to be set explicitly for each network host, you must also determine the corresponding Ethernet address for each DNA address. For reference information on the relation of the Ethernet address and the DNA address: See the section "DNA Requirement for Ethernet Address", page 9.

If the node number of any host exceeds the MAX ADDRESS parameter of the NCP (usually 32), you should change MAX ADDRESS to a higher number: See the section "Setting the MAX ADDRESS Parameter of the VAX/VMS NCP", page 11.

1. Assign a DNA address to each host.

2. Type the following function at a Lisp Listener:

   ```
   (DEFUN DNA-TO-ETHERNET-ADDRESS(ADDR)
     (LET ((ADDRESS (NETI:PARSE-DNA-ADDRESSADDR)))
          (FORMAT T "~& Ethernet Address: AA-00-04-00--~16,2,48R--~16,2,48R
             " (LDB #O0010 ADDRESS) (LDB #O1010 ADDRESS))))
   ```

3. Use the function to determine the Ethernet address for each DNA address assigned. In this example, the DNA address of the host is 1.94.

   ```
   (dna:dna-to-ethernet-address"1.94")
   ```

3.3 Using a VAX/VMS Host as a SYS Host

The DNA software package enables a VAX/VMS to act as a SYS host for the site.

Any new sites that intend to use the VAX/VMS as a SYS host need to follow the installation instructions in this section. The problem is a bootstrapping one: in order to install the Symbolics Release 6.1 software on the VAX, you need to have

DNA software installed and operational. However, you must already have Symbolics Release 6.1 software installed before you can install DNA software. The solution to this problem is to do a preliminary installation using a Symbolics computer as a SYS host, bring up DNA on that computer, and then reinstall the software on the VAX/VMS host.

Sites that have the VAX/VMS Chaosnet software running need not concern themselves with this problem; they can install Symbolics Release 6.1 on the VAX using Chaosnet, and then install DNA. This holds true for sites running DNA Release 1 software as well.

If you intend to use the VAX as a SYS host, and your site is not currently running VAX/VMS Chaosnet software or DNA Release 1 software, follow these steps:

1. Create a sys.translations file indicating that a chosen Symbolics computer is the SYS host.

2. Install the Symbolics Release 6.1 software on that Symbolics computer, following instructions in the *Software Installation Guide.*

3. Install the DNA software on that Symbolics computer, following the instructions in this document. Once you are finished, and the Symbolics computer can communciate with the VAX using DNA, continue to the next step in this section.

4. Edit the sys.translations file to indicate that the VAX/VMS host is acting as a SYS host.

5. Reinstall the Symbolics Release 6.1 software from the distribution tape.

6. Reinstall the DNA software from the distribution tape.

3.4 Installing DNA on the First Symbolics Computer

You must have completely installed Release 6.1 before beginning the installation of DNA. The order of the installation steps is essential.

To install DNA from the distribution tape, perform the following steps in the numbered order:

1. Edit the first Symbolics computer's boot file.

   To use the DNA network protocols, each Symbolics computer must have a 48-bit Ethernet address set in its boot file. That Ethernet address is derived from the machine's assigned DNA address. Add the command to set the Ethernet address to the first Symbolics computer's boot file. The line resembles the following example, with xx and yy replaced with appropriate values:

```
Set Ethernet-address AA-00-04-00-xx-yy
```

A host with DNA address 3.7 would have this line in the boot file:

```
Set Ethernet-address AA-00-04-00-07-0C
```

2. Cold boot the Symbolics computer.

   Once you have added the command to the boot file that sets the host's Ethernet address, you must cold boot the machine in order to make the Ethernet address take effect.

3. Place the distribution tape in the cartridge tape drive.

4. Create a system location file for DNA.

   To allow the system to automatically find the DNA system files, you must create a system location file in your site directory (sys:site;) named dna.system. It must contain the following text:

   ```
   ;;; -*- Mode: Lisp; -*-

   (SI:SET-SYSTEM-SOURCE-FILE"DNA" "SYS:DNA;SYSTEM")
   ```

   This indicates to the system that the DNA system is stored in the logical directory DNA on the logical host SYS, and that the system declaration file is in sys:dna;system.lisp.

5. Load the distribution tape by typing the following form to a Lisp Listener:

   ```
   (dis:load-distribution-tape)
   ```

   The distribution loader asks you whether the tape is loaded on the local machine or on another host on the Ethernet:

   ```
   Type host or spec for Distribution tape
   (default (Return) = Local: Cart, reclen = 4096):
   ```

   If the tape is loaded on the local machine, press RETURN . If not, you can press HELP for a list of choices. See the section "Tape Specs" in *Installation and Site Operations.*

   Once you have answered that first quesion, a menu of `Items to be loaded` appears. Use [Do It] to restore the DNA files from the tape.

   If you interrupt the restoration process before it completes and later attempt to restore the files on the tape again, the function **dis:load-distribution-tape** displays a message stating that the file or files already exist with the correct author and date, and skips those files.

6. Remove the tape from the cartridge tape drive.

7. Enable DNA on the Symbolics computer.

   Give the following command to a Lisp Listener:

   ```
   load system dna :automatic answer
   ```

3.5 Installing Server Programs on VAX/VMS

This section describes how to install the server programs on VAX/VMS. The installation procedure loads 16 new files, totalling approximately 300 blocks, into the SYS$SYSTEM directory.

1. Begin by logging in to the VAX/VMS SYSTEM account.

2. Load the tape on a tape drive accessible to the VAX.

3. Mount the tape with the following VAX/VMS DCL command, replacing c with the controller ID, and n with the unit number:

   ```
   $ MOUNT/FOREIGN MTcn:
   ```

   One example of this command is $ MOUNT/FOREIGN MTA0:.

4. Restore the server images from the tape, again replacing c with the controller ID, and n with the unit number:

   ```
   $ BACKUP/LOG MTcn:VMSDNA1.BCK/SAVE/SELECT=*.*SYS$SYSTEM:*.*/VERIFY
   ```

5. To enable the servers when DECnet is running on the VAX/VMS system, invoke the command procedure SYS$SYSTEM:LISPMDNA.COM with the following command:

   ```
   $ @SYS$SYSTEM:LISPMDNA.COM
   ```

   This command can be added to the site-specific start-up command file (usually SYS$MANAGER:SYSTARTUP.COM) after the invocation of the DECnet startup command file SYS$MANAGER:STARTNET.COM. Alternatively, the server objects can be defined permanently in the site network database. See the sections in VAX/VMS *Guide to Networking on VAX/VMS* (published by Digital Equipment Corporation) that discuss the concept and mechanics of defining "Objects" in the network database.

3.6 Editing the VAX/VMS DECnet Database

The VAX/VMS system does not consult the namespace database for information on networks. VAX/VMS has its own database that maps the names of DNA hosts to their DNA addresses. You can add the Symbolics Lisp Machines that are now DNA hosts to the VAX/VMS database. The node addresses must be the same in the namespace database and the VAX/VMS DECnet database. See the *VAX/VMS Guide to Networking*, published by Digital Equipment Corporation, for further information.

Strictly speaking, it is not necessary to add the Symbolics Lisp Machines to the VAX/VMS DECnet database. In general, the DNA protocols are used to provide the Symbolics Lisp Machine user with a service on the VAX/VMS system, such as file transfer. Therefore, the Symbolics computers must have information about any VAX/VMS systems on the network, their addresses, and the services they provide. Symbolics Lisp Machines consult the namespace database for this information. However, the VAX needs to know about a Symbolics computer only if a VAX user wants to invoke a service on a Symbolics computer. This would be an unusual situation, but if it occurs the VAX would need to have up-to-date information in its VAX/VMS DECnet database to make the connection.

It is good practice to keep both databases (the namespace database and the VAX/VMS DECnet database) current.

3.7 Updating the Namespace Database for DNA

This section describes the changes to be made to the namespace database. This section is for new DNA sites only. Existing DNA sites should already have the namespace database set up. If you are not yet familar with the namespace editor:

> See the section "Updating the Namespace Database" in *User's Guide to Symbolics Computers.*
> See the section "Namespace System" in *Networks.*

1. Enter the namespace editor by choosing it from the System Menu or by giving the Edit Namespace Object command.

2. Create a new network object for DNA in the namespace database.

   Use [Create] to create a new object. The class of the object to create is Network. The name of the network object should be DNA. By clicking on the fields of the record, you can add the site name and optional nicknames (DNA and DECNET are appropriate). The final record should be similar to this:

```
Type*: DNA
Site: YOUR-SITE
Nickname: DNA
Nickname: DECNET
Nickname: Name
Subnet: Pair: Token Set: Global-name Token
User Property: Pair: Global-name Token
```

3. Add the DNA host address to the host object for each host (whether VAX/VMS or Symbolics computer) that will be using DNA:

   Find a line like:

   ```
   Address: Pair: Network Token
   ```

   Click on *Network*, and change the value to DNA. Click on *Token*, and change the value to be the DNA address of the host. A host with DNA address 3.7 should then have an entry like:

   ```
   Address: Pair: DNA 3.7
   ```

4. For each VAX/VMS host that runs DNA, add the service entries to the host object. Find a line like:

   Service: **Set:** *Global-name*

   Click left on **Set:** to add the following sets. The resulting lines appear like:

   ```
   Service: Set: MAIL-TO-USER DNA DNA-MAIL Global-name
   Service: Set: FILE DNA DAP Global-name
   Service: Set: TIME DNA DNA-LMTIME Global-name
   Service: Set: UPTIME DNA DNA-LMUPTIME Global-name
   Service: Set: LOGIN DNA CTERM Global-name
   Service: Set: TAPE DNA RTAPE Global-name
   Service: Set: SHOW-USERS DNA ASCII-NAME Global-name
   Service: Set: LOOPBACK DNA DNA-LOOPBACK-MIRRORGlobal-name
   ```

5. For each Symbolics computer host that will be running DNA, add the following service entries to the host object:

   ```
   Service: Set: LOOPBACK DNA DNA-LOOPBACK-MIRRORGlobal-name
   Service: Set: MAIL-TO-USER DNA DNA-MAIL Global-name
   Service: Set: FILE DNA DAP Global-name
   ```

6. Verify DNA host addresses in the namespace database.

   Once you have finished editing the namespace database, you can verify that the host objects now have a DNA address. The function

dna:list-all-host-addresses queries the namespace database for all hosts with DNA addressses, and prints the list on your screen. Be sure that each host has an unique DNA address.

```
(dna:list-all-host-addresses)
```

7. Edit User Objects.

During file transfer, it is convenient for you if the Symbolics computer knows what your home directory is on the foreign host machine. That way you can specify a partial pathname, and the Symbolics computer will be able to translate that incomplete pathname into a full pathname. The Symbolics computer "knows" this information if it is stored in your user object in the namespace database.

Edit the user object for each user who will be using DNA for file transfer. Find an entry for User Property. Substitute for *Global-name* the word HOMEDIR. Substitute for *Token* the directory component of your home directory on the foreign host that you will most often use for file transfer.

For a user whose home directory on VAX/VMS is [jones], the entry in the user object looks like:

`User Property:` **Pair:** `HOMEDIR jones` *Global-name*

Only one home directory can be stored in the user object. If you should later want to do a file transfer to another host (in which your home directory is different than the one stored in your user object), you must specify the complete pathname of the file to be transferred.

3.8 Resetting and Enabling the Network for DNA

After installing DNA and configuring the site, it is necessary to reset the local network interface, and then enable it again. Give the following command to a Lisp Listener:

```
reset network
```

3.9 Verifying the DNA Installation

You can run two loopback tests from a Symbolics computer to verify the DNA installation. The local loopback sends traffic through the Symbolics computer's network interface and checks that the traffic is received intact. The remote loopback sends traffic to a foreign host and checks that the traffic is received intact. Both loopbacks should run successfully if the DNA installation is complete.

At a Lisp Listener, type:

```
(dna:verify-installationhostname)
```

First substitute your local host's name and try the local loopback. The status line displays the DNA network activity as it happens. If the loopback test completes successfully, the last words displayed are "Installation successful".

Second, substitute the name of another host running DNA to do the remote loopback. If the loopback test is successful, you know that traffic can flow over DNA protocols to another DNA host.

3.10 Installing DNA for Entire Site

Once the site is configured, there are two ways to install DNA on other machines at the site:

1. Loading DNA individually on each Symbolics computer.

2. Saving a world containing DNA and copying that world to other Symbolics computers.

This section describes both ways. In either case, each Symbolics computer running DNA must be configured properly, by having its Ethernet address set in its boot file.

3.10.1 Configuring Each Symbolics Computer for DNA

In the boot file of every Symbolics computer that will be using DNA, add the command to set the Ethernet address for that host. The command line resembles the following example, with xx and yy replaced with appropriate values:

```
Set Ethernet-address AA-00-04-00-xx-yy
```

A host with DNA address 3.7 would have this line in the boot file:

```
Set Ethernet-address AA-00-04-00-07-0C
```

Cold boot each Symbolics computer after editing the boot file, in order to set the Ethernet address for that machine.

3.10.2 Loading DNA on a Symbolics Computer

To load DNA on an individual Symbolics computer, type the following command at a Lisp Listener:

```
load system dna :automatic-answer
```

DNA is now loaded. You can save the system by using the Save World command at this point if you do not wish to reload the software each time you boot the machine.

If you do not immediately save the world, you must give the following command before using DNA:

```
Reset Network
```

3.10.3 Saving and Distributing the DNA World

Once you have installed DNA on one Symbolics computer and verified the installation, you can use the Save World command to save the world containing DNA. We recommend that you garbage collect before saving the world.

You can share the saved world containing DNA among the other Symbolics Lisp machines at the site by using the Copy World command.

For further information on the commands mentioned in this section:

> See the section "Turning on the Garbage Collector" in *User's Guide to Symbolics Computers.*
> See the section "Save World Command" in *User's Guide to Symbolics Computers.*
> See the section "Copy World Command" in *User's Guide to Symbolics Computers.*

Index

2

Release Notes for DNA Software Package Release 2.0 1

6

New Logical Pathname Translation for VAX/VMS V4 Hosts in Release 6.1 2

A

Resetting File Access Paths 16
Using direct access with DNA 1
Adding New Applications That Use DNA 18
DNA Requirement for Ethernet Address 9
Ethernet address 32
Mapping a Chaos Address Into a DNA Address 8
Assigning DNA and Ethernet Addresses 28
Format of DNA Addresses 7
DNA addressing 32
DNA Network Addressing 6
Summary of DNA Addressing Requirements 6
Mapping a Chaos Address Into a DNA Address 8
MAX ADDRESS parameter 6
Setting the MAX ADDRESS Parameter of the VAX/VMS NCP 11
Address Resolution Protocol 9
Writing a New DNA Application for the Symbolics Computer 18
Writing a New DNA Application for the VAX 21
Adding New Applications That Use DNA 18
DNA area number 7, 8
Assigning DNA and Ethernet Addresses 28

B

Boot file 29

C

DNA password cache 15
Mapping a Chaos Address Into a DNA Address 8
Using Peek to Check DNA Status 15
`NETWORK X` command 12
Reset Network Command 16
DNA COMMAND server 18
Installing DNA on the First Symbolics Computer 29
Loading DNA on a Symbolics Computer 35
Using DNA on the Symbolics Computer 11
Writing a New DNA Application for the Symbolics Computer 18
Configuring Each Symbolics Computer for DNA 35
Configuring Each Symbolics Computer for DNA 35
Connecting to a Remote Host Over the Network 12
Hint for Connecting to Remote Hosts 13
DNA Security Considerations 14

D D D

Editing the VAX/VMS DECnet Database 32
Namespace database 32
Updating the Namespace Database for DNA 32
DNA Debugging Tools 15
Editing the VAX/VMS DECnet Database 32
References to DECnet Protocol Specifications 16
Desirability of Network Protocols 11
Using direct access with DNA 1
Home directory 32
dna: **disable-password-caching** function 15
Saving and Distributing the DNA World 36
Adding New Applications That Use DNA 18
Configuring Each Symbolics Computer for DNA 35
Hardware and Software Requirements of DNA 6
Loading DNA 29
Reference Information on DNA 5
Resetting and Enabling the Network for DNA 34
Updating the Namespace Database for DNA 32
Using direct access with DNA 1
DNA addressing 32
DNA Installation 27
DNA password 14
DNA Services 6
dna:disable-password-caching function 15
dna:enable-password-caching function 14
dna:flush-password-cache function 15
Mapping a Chaos Address Into a DNA Address 8
Format of DNA Addresses 7
Summary of DNA Addressing Requirements 6
Assigning DNA and Ethernet Addresses 28
Writing a New DNA Application for the Symbolics Computer 18
Writing a New DNA Application for the VAX 21
DNA area number 7, 8
DNA COMMAND server 18
DNA Debugging Tools 15
DNA EVAL server 21
Installing DNA for Entire Site 35
DNA Implementation Notes 17
Overview of DNA Installation 27
Verifying the DNA Installation 34
DNA loopback test 34
DNA Network Addressing 6
DNA network object 29
DNA network services 32
DNA node number 7, 8
Loading DNA on a Symbolics Computer 35
Installing DNA on the First Symbolics Computer 29
Using DNA on the Symbolics Computer 11
DNA password cache 15
DNA Requirement for Ethernet Address 9
DNA Security Considerations 14
Overview of DNA Software Package 5
Release Notes for DNA Software Package Release 2.0 1
Using Peek to Check DNA Status 15
DNA system location file 29
Saving and Distributing the DNA World 36
Double login 14

E E E

Configuring Each Symbolics Computer for DNA 35
Editing the VAX/VMS DECnet Database 32
dna: enable-password-caching function 14
Resetting and Enabling the Network 16
Resetting and Enabling the Network for DNA 34
Endnode implementation 17
ENDNODE-HELLOS 15
Installing DNA for Entire Site 35
Service entries 32
Ethernet address 32
Ethernet interface 6
Set Ethernet-Address 9, 29
DNA Requirement for Ethernet Address 9
Assigning DNA and Ethernet Addresses 28
DNA EVAL server 21

F F F

Boot file 29
DNA system location file 29
File transfer 32
Resetting File Access Paths 16
Installing DNA on the First Symbolics Computer 29
dna: flush-password-cache function 15
Format of DNA Addresses 7
dna:disable-password-caching function 15
dna:enable-password-caching function 14
dna:flush-password-cache function 15

H H H

Hardware and Software Requirements of DNA 6
Hellos 15
Hint for Connecting to Remote Hosts 13
Home directory 32
Using a VAX/VMS Host as a SYS Host 28
Using a VAX/VMS Host as a SYS Host 28
Connecting to a Remote Host Over the Network 12
Hint for Connecting to Remote Hosts 13
New Logical Pathname Translation for VAX/VMS V4 Hosts in Release 6.1 2

I I I

Endnode implementation 17
DNA Implementation Notes 17
Reference Information on DNA 5
DNA Installation 27
Overview of DNA Installation 27
Verifying the DNA Installation 34
Installing DNA for Entire Site 35
Installing DNA on the First Symbolics Computer 29
Installing Server Programs on VAX/VMS 31
Ethernet interface 6
Mapping a Chaos Address Into a DNA Address 8
ISO 17

K

NETWORK Key 13
SELECT T key 12

L

Loading DNA 29
Loading DNA on a Symbolics Computer 35
DNA system location file 29
New Logical Pathname Translation for VAX/VMS V4 Hosts in Release 6.1 2
Double login 14
LOGIN service 12
DNA loopback test 34

M

Mapping a Chaos Address Into a DNA Address 8
MAX ADDRESS parameter 6
Setting the MAX ADDRESS Parameter of the VAX/VMS NCP 11
Overstrike network mode 13

N

Namespace database 32
Updating the Namespace Database for DNA 32
Setting the MAX ADDRESS Parameter of the VAX/VMS NCP 11
Connecting to a Remote Host Over the Network 12
Resetting and Enabling the Network 16
NETWORK Key 13
DNA Network Addressing 6
Reset Network Command 16
Resetting and Enabling the Network for DNA 34
Overstrike network mode 13
DNA network object 29
Recovering From a Network Problem 16
Desirability of Network Protocols 11
DNA network services 32
NETWORK X command 12
Adding New Applications That Use DNA 18
Writing a New DNA Application for the Symbolics Computer 18
Writing a New DNA Application for the VAX 21
New Logical Pathname Translation for VAX/VMS V4 Hosts in Release 6.1 2
DNA node number 7, 8
DNA Implementation Notes 17
Release Notes for DNA Software Package Release 2.0 1
DNA area number 7, 8
DNA node number 7, 8
telnet: ***nvt-overstrike-default*** 13

O

DNA network object 29
OSI 17
Overstrike network mode 13
Connecting to a Remote Host Over the Network 12
Overview of DNA Installation 27
Overview of DNA Software Package 5

P P P

Overview of DNA Software	Package 5
Release Notes for DNA Software	Package Release 2.0 1
MAX ADDRESS	parameter 6
Setting the MAX ADDRESS	Parameter of the VAX/VMS NCP 11
	Password 11
DNA	password 14
DNA	password cache 15
New Logical	Pathname Translation for VAX/VMS V4 Hosts in Release 6.1 2
Resetting File Access	Paths 16
Using	Peek to Check DNA Status 15
Recovering From a Network	Problem 16
Supdup	program 12
Telnet	program 12
Using the Terminal	Program 12
Installing Server	Programs on VAX/VMS 31
Address Resolution	Protocol 9
Desirability of Network	Protocols 11
References to DECnet	Protocol Specifications 16

R R R

	Recovering From a Network Problem 16
	Reference Information on DNA 5
	References to DECnet Protocol Specifications 16
Release Notes for DNA Software Package	Release 2.0 1
New Logical Pathname Translation for VAX/VMS V4 Hosts in	Release 6.1 2
	Release Notes for DNA Software Package Release 2.0 1
Connecting to a	Remote Host Over the Network 12
Hint for Connecting to	Remote Hosts 13
DNA	Requirement for Ethernet Address 9
Summary of DNA Addressing	Requirements 6
Hardware and Software	Requirements of DNA 6
	Reset Network Command 16
	Resetting and Enabling the Network 16
	Resetting and Enabling the Network for DNA 34
	Resetting File Access Paths 16
Address	Resolution Protocol 9
	ROUTER-HELLOS 15

S S S

	Saving and Distributing the DNA World 36
DNA	Security Considerations 14
	SELECT T key 12
DNA COMMAND	server 18
DNA EVAL	server 21
Installing	Server Programs on VAX/VMS 31
LOGIN	service 12
	Service entries 32
DNA	Services 6
DNA network	services 32
	Set Ethernet-Address 9, 29
	Setting the MAX ADDRESS Parameter of the VAX/VMS NCP 11
Installing DNA for Entire	Site 35
Overview of DNA	Software Package 5
Release Notes for DNA	Software Package Release 2.0 1
Hardware and	Software Requirements of DNA 6
References to DECnet Protocol	Specifications 16

Using Peek to Check DNA	Status 15
	Summary of DNA Addressing Requirements 6
	Supdup program 12
Installing DNA on the First	Symbolics Computer 29
Loading DNA on a	Symbolics Computer 35
Using DNA on the	Symbolics Computer 11
Writing a New DNA Application for the	Symbolics Computer 18
Configuring Each	Symbolics Computer for DNA 35
Using a VAX/VMS Host as a	SYS Host 28
DNA	system location file 29

T

	Telnet program 12
	telnet:*nvt-overstrike-default* 13
Using the	Terminal Program 12
DNA loopback	test 34
Adding New Applications	That Use DNA 18
SELECT	T key 12
DNA Debugging	Tools 15
File	transfer 32
New Logical Pathname	Translation for VAX/VMS V4 Hosts in Release 6.1 2

U

	Updating the Namespace Database for DNA 32
	Upgrading VAX/VMS V3 to V4 2
Adding New Applications That	Use DNA 18
	Using a VAX/VMS Host as a SYS Host 28
	Using direct access with DNA 1
	Using DNA on the Symbolics Computer 11
	Using Peek to Check DNA Status 15
	Using the Terminal Program 12

V

Upgrading VAX/VMS	V3 to V4 2
Upgrading VAX/VMS V3 to	V4 2
New Logical Pathname Translation for VAX/VMS	V4 Hosts in Release 6.1 2
Writing a New DNA Application for the	VAX 21
Installing Server Programs on	VAX/VMS 31
Editing the	VAX/VMS DECnet Database 32
Using a	VAX/VMS Host as a SYS Host 28
Setting the MAX ADDRESS Parameter of the	VAX/VMS NCP 11
Upgrading	VAX/VMS V3 to V4 2
New Logical Pathname Translation for	VAX/VMS V4 Hosts in Release 6.1 2
	Verifying the DNA Installation 34
VMS	version 6
	VMS version 6

W

Saving and Distributing the DNA	World 36
	Writing a New DNA Application for the Symbolics Computer 18
	Writing a New DNA Application for the VAX 21

X X X

NETWORK X command 12