J 6

```
NNN         NNN  MMM         MMM  LLL
NNN         NNN  MMM         MMM  LLL
NNN         NNN  MMM         MMM  LLL
NNN         NNN  MMMMMM   MMMMMM  LLL
NNN         NNN  MMMMMM   MMMMMM  LLL
NNN         NNN  MMMMMM   MMMMMM  LLL
NNNNNN      NNN  MMM   MMM   MMM  LLL
NNNNNN      NNN  MMM   MMM   MMM  LLL
NNNNNN      NNN  MMM   MMM   MMM  LLL
NNN   NNN   NNN  MMM         MMM  LLL
NNN   NNN   NNN  MMM         MMM  LLL
NNN   NNN   NNN  MMM         MMM  LLL
NNN      NNNNNN  MMM         MMM  LLL
NNN      NNNNNN  MMM         MMM  LLL
NNN      NNNNNN  MMM         MMM  LLL
NNN         NNN  MMM         MMM  LLL
NNN         NNN  MMM         MMM  LLL
NNN         NNN  MMM         MMM  LLL
NNN         NNN  MMM         MMM  LLLLLLLLLLLLLLL
NNN         NNN  MMM         MMM  LLLLLLLLLLLLLLL
NNN         NNN  MMM         MMM  LLLLLLLLLLLLLLL
```

```
_$
Ps
--
NP
NP
$G
$O
NP
PA
_L
```

E 13

```
**FILE**ID**NMLMAIN
```

```
NML
V04
```

```
NN      NN  MM      MM  LL          MM      MM    AAAAAA      IIIIII    NN      NN
NN      NN  MM      MM  LL          MM      MM    AAAAAA      IIIIII    NN      NN
NN      NN  MMMM  MMMM  LL          MMMM  MMMM  AA      AA      II      NN      NN
NN      NN  MMMM  MMMM  LL          MMMM  MMMM  AA      AA      II      NN      NN
NNNN    NN  MM  MM  MM  LL          MM  MM  MM  AA      AA      II      NNNN    NN
NNNN    NN  MM  MM  MM  LL          MM  MM  MM  AA      AA      II      NNNN    NN
NN  NN  NN  MM      MM  LL          MM      MM  AA      AA      II      NN  NN  NN
NN  NN  NN  MM      MM  LL          MM      MM  AA      AA      II      NN  NN  NN
NN    NNNN  MM      MM  LL          MM      MM  AAAAAAAAAA      II      NN    NNNN
NN    NNNN  MM      MM  LL          MM      MM  AAAAAAAAAA      II      NN    NNNN
NN      NN  MM      MM  LL          MM      MM  AA      AA      II      NN      NN      ....
NN      NN  MM      MM  LL          MM      MM  AA      AA      II      NN      NN      ....
NN      NN  MM      MM  LLLLLLLLLL  MM      MM  AA      AA    IIIIII    NN      NN      ....
NN      NN  MM      MM  LLLLLLLLLL  MM      MM  AA      AA    IIIIII    NN      NN      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

;

```
0001  0 %TITLE 'Network Management Listener main module'
0002  0 MODULE NML$MAIN (MAIN = NML$MAIN,
0003  0                 ADDRESSING_MODE (NONEXTERNAL=GENERAL),
0004  0                 ADDRESSING_MODE (EXTERNAL=GENERAL),
0005  0                 IDENT = 'V04-000') =
0006  0
0007  1 BEGIN
0008  1
0009  1 !
0010  1 !****************************************************************************
0011  1 !*                                                                          *
0012  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0013  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0014  1 !*  ALL RIGHTS RESERVED.                                                    *
0015  1 !*                                                                          *
0016  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0017  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0018  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0019  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0020  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0021  1 !*  TRANSFERRED.                                                            *
0022  1 !*                                                                          *
0023  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0024  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0025  1 !*  CORPORATION.                                                            *
0026  1 !*                                                                          *
0027  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0028  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0029  1 !*                                                                          *
0030  1 !*                                                                          *
0031  1 !****************************************************************************
0032  1 !
0033  1
0034  1 !++
0035  1 ! FACILITY:  DECnet-VAX V2.0 Network Management Listener
0036  1 !
0037  1 ! ABSTRACT:
0038  1 !
0039  1 !       This is the main module for NML.
0040  1 !
0041  1 ! ENVIRONMENT:  VAX/VMS Operating System
0042  1 !
0043  1 ! AUTHOR:  Distributed Systems Software Engineering
0044  1 !
0045  1 ! CREATION DATE:  1-OCT-1979
0046  1 !
0047  1 ! MODIFIED BY:
0048  1 !
0049  1 !       V03-004 MKP0005         Kathy Perko     3-Jan-1984
0050  1 !               Convert old $TRNLOG to new $TRNLNM system service for
0051  1 !               translating logical names.
0052  1 !
0053  1 !       V03-003 MKP0004         Kathy Perko     21-April-1983
0054  1 !               Delete service functions from NML.
0055  1 !
0056  1 !       V03-002 MKP0003         Kathy Perko     1-March-1983
0057  1 !               Get rid of extra NET: channel to NETACP which was used
```

```
1 !                       when NML was part the network booting process.
1 !
1 !               V03-001 MKP0002         Kathy Perko     17-Nov-1982
1 !                       Add checking for Network Management Version 4.0.0 and
1 !                       consolidate the version checking code into one routine.
1 !
1 !               V005    MKP0001         Kathy Perko     17-Dec-1981
1 !                       Bypass version number checking for autoservice functions.
1 !
1 !               V004    TMH0004         Tim Halvorsen   12-Oct-1981
1 !                       Store NICE version number of NCP we are communicating
1 !                       with, so that some level of V2.0 compatibility can be done.
1 !                       Mark Phase II (missing version number) as V1.3.0 to
1 !                       NML$INITIALIZE routine.
1 !
1 !               V003    MKP0003         Kathy Perko     03-Sep-1981
1 !                       Temporarily tell 2.0 NCP's that NML is 2.0 also.
1 !
1 !               V002    TMH0002         Tim Halvorsen   31-Aug-1981
1 !                       Allow communication with 2.0.0 NCPs.
1 !
1 !               V001    TMH0001         Tim Halvorsen   20-Jul-1981
1 !                       Use sharable image callable interface.
1 !                       Make program exit quietly when the link to NCP is aborted.
1 !                       Remove network initialization code, since NCP no longer
1 !                       initiates NML as a detached process during startup.
1 !--
1
```

```
1 %SBTTL 'Declarations'
1
1 !
1 ! TABLE OF CONTENTS:
1 !
1
1 FORWARD ROUTINE
1     NML$MAIN             : NOVALUE,
1     NML_COMMANDS         : NOVALUE,
1     NML_INITIALIZE,
1     NML_CONNECT,
1     NML_VALIDLINK,
1     NML_GETMODE          : NOVALUE,
1     NML_RECEIVE,                          ! Receive NICE message over link
1     NML_RESPONSE:       NOVALUE;          ! Send message response over link
1
1 !
1 ! INCLUDE FILES:
1 !
1
1 LIBRARY 'LIB$:NMLLIB';                    ! Facility-wide definitions
1 LIBRARY 'SHRLIB$:NMALIBRY';               ! NICE definitions
1 LIBRARY 'SHRLIB$:NET';                    ! NETACP QIO interface
1 LIBRARY 'SYS$LIBRARY:STARLET';            ! VMS common definitions
1
1 !
1 ! NML mode symbols (set in NML$B_MODE)
1 !
1
1 LITERAL
1     NML$_INIT    = 0,   ! Initializing
1     NML$_CONNECT = 1;   ! Started via connect
1
1 !
1 ! OWN STORAGE:
1 !
1 BIND
1     NML_VERSION = UPLIT BYTE (NML$K_VERSION,
1                               NML$K_DEC_ECO,
1                               NML$K_USER_ECO);
1
1 OWN
1     NML$B_MODE       : BYTE
1                        INITIAL (NML$_INIT), ! NML mode of operation
1     NML$L_CMDNCBLEN,                      ! Length of NCB data
1     NML$W_CMD_CHAN: WORD,                 ! Channel to logical link to NCP
1     NML$A_NCBADDRESS,                     ! Address of NCB
1     NML$A_NCBDATA,                        ! Address of optional NCB data field
1     NML$Q_SRVIDDSC: BLOCK [8,BYTE],       ! Descriptor of autoservice NCB
1     NML$AB_RCVBUFFER: VECTOR [nml$k_rcvbflen, BYTE];    ! Message buffer
1
1 !
1 ! EXTERNAL REFERENCES:
1 !
1
1 EXTERNAL ROUTINE
1     NML$INITIALIZE,                       ! Initialize NICE processor
```

```
:   144          0143  1      NML$PROCESS_NICE:    NOVALUE,          ! Process a single NICE message
:   145          0144  1      NML$TERMINATE:       NOVALUE,          ! Terminate NICE processor
:   146          0145  1      LIB$ASN_WTH_MBX;                       ! Assign channel with mailbox
```

```
;   148          0146  1 %SBTTL 'NML$MAIN  Main routine'
;   149          0147  1 ROUTINE NML$MAIN : NOVALUE =
;   150          0148  1
;   151          0149  1 !++
;   152          0150  1 ! FUNCTIONAL DESCRIPTION:
;   153          0151  1 !
;   154          0152  1 !       This is the main routine for the DECnet-VAX Network Management
;   155          0153  1 !       Listener (NML).
;   156          0154  1 !
;   157          0155  1 ! ROUTINE VALUE:
;   158          0156  1 ! COMPLETION CODES:
;   159          0157  1 !
;   160          0158  1 !       Always returns NML$_STS_SUC.
;   161          0159  1 !
;   162          0160  1 !--
;   163          0161  1
;   164          0162  2 BEGIN
;   165          0163  2
;   166          0164  2 !
;   167          0165  2 ! Determine how we were initiated.
;   168          0166  2 !
;   169          0167  2 IF NML_INITIALIZE () THEN
;   170          0168  3     BEGIN
;   171          0169  3     NML_COMMANDS ();                ! NICE command mode
;   172          0170  3     NML$TERMINATE();                ! Terminate NICE processor
;   173          0171  2     END;
;   174          0172  2
;   175          0173  2 $EXIT (CODE = TRUE);
;   176          0174  2
;   177          0175  1 END;                       ! End of NML


                                                                   .TITLE  NML$MAIN Network Management Listener main modul
;                                                                                                                          e
                                                                   .IDENT  \V04-000\

                                                                   .PSECT  $PLIT$,NOWRT,NOEXE,2

                                            00  00  04  00000 P.AAA:  .BYTE   4, 0, 0                                   ;

                                                                   .PSECT  $OWN$,NOEXE,2

                                                    00  00000 NML$B_MODE:
                                                                   .BYTE   0                                         ;
                                                        00001      .BLKB   3
                                                        00004 NML$L_CMDNCBLEN:
                                                                   .BLKB   4
                                                        00008 NML$W_CMD_CHAN:
                                                                   .BLKB   2
                                                        0000A      .BLKB   2
                                                        0000C NML$A_NCBADDRESS:
                                                                   .BLKB   4
                                                        00010 NML$A_NCBDATA:
                                                                   .BLKB   4
                                                        00014 NML$Q_SRVIDDSC:
                                                                   .BLKB   8
                                                        0001C NML$AB_RCVBUFFER:
```

```
                                                                        .BLKB   512

                                                                NML_VERSION=       P.AAA
                                                                        .EXTRN  NML$INITIALIZE, NML$PROCESS_NICE
                                                                        .EXTRN  NML$TERMINATE, LIB$ASN_WTH_MBX
                                                                        .EXTRN  SYS$EXIT

                                                                        .PSECT  $CODE$,NOWRT,2

                                                       0000 00000 NML$MAIN:
                                                                        .WORD   Save nothing                                      ; 0147
                        00000000V  00              00  FB 00002         CALLS   #0, NML_INITIALIZE                                ; 0167
                                   0E              50  E9 00009         BLBC    R0, 1$                                            ;
                        00000000V  00              00  FB 0000C         CALLS   #0, NML_COMMANDS                                  ; 0169
                        00000000G  00              00  FB 00013         CALLS   #0, NML$TERMINATE                                 ; 0170
                                                   01  DD 0001A 1$:     PUSHL   #1                                                ; 0173
                        00000000G  00              01  FB 0001C         CALLS   #1, SYS$EXIT                                      ;
                                                       04 00023         RET                                                       ; 0175

; Routine Size:  36 bytes,    Routine Base:  $CODE$ + 0000
```

```
1 %SBTTL 'NML_COMMANDS  Main command processing routine'
1 ROUTINE NML_COMMANDS: NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine is the main command processing routine.  NICE messages
1 !       are parsed to determine the requested function and then the proper
1 !       routine is called to perform the function.
1 !
1 !--
1
2 BEGIN
2
2 LOCAL
2     desc:       VECTOR [2],              ! Message descriptor
2     status;                              ! Temporary status
2
2 !
2 ! Receive NICE messages and perform valid functions as long as the link
2 ! is active.
2 !
2
2 WHILE TRUE
2 DO
3     BEGIN
3     desc [1] = nml$ab_rcvbuffer;
3     status = nml_receive (.desc [1],
3                           nml$k_rcvbflen,
3                           desc [0]);
3     IF .status
3     THEN
3         nml$process_nice(desc, nml_response)
3
3 !
3 ! If the message received was too large for the buffer then send a status
3 ! message indicating the error.  Any other errors indicate that the link
3 ! has been broken and NML should exit.
3 !
3
3     ELSE
4         BEGIN
4         IF .STATUS EQLU SS$_LINKABORT   ! If normal exit (via link DASSGN
4             OR .STATUS EQLU SS$_LINKDISCON      ! or formal disconnect)
4         THEN
4             RETURN;                      ! then exit successfully
4
4         IF .STATUS EQLU SS$_DATAOVERUN  ! If message was too large,
4         THEN
4             NML_RESPONSE(UPLIT(1, UPLIT BYTE(NMA$C_STS_SIZ))) ! Send size error
4         ELSE
4             SIGNAL_STOP (.STATUS);      ! Signal fatal error
3         END;
3
2     END;
2
1 END;
```

```
                                              .PSECT  $PLIT$,NOWRT,NOEXE,2

                            FC  00003 P.AAC:  .BYTE   -4                                          ;
                      00000001  00004 P.AAB:  .LONG   1                                           ;
                      00000000' 00008         .ADDRESS P.AAC                                      ;


                                              .PSECT  $CODE$,NOWRT,2

                           000C 00000 NML_COMMANDS:
                                              .WORD   Save R2,R3                                  ; 0177
           53 00000000V  00  9E 00002         MOVAB   NML_RESPONSE, R3                            ;
           5E            08  C2 00009         SUBL2   #8, SP                                      ;
      04   AE 00000000'  00  9E 0000C 1$:     MOVAB   NML$AB_RCVBUFFER, DESC+4                    ; 0202
                         5E  DD 00014         PUSHL   SP                                          ; 0205
           7E     0200   8F  3C 00016         MOVZWL  #512, -(SP)                                 ; 0203
                    0C   AE  DD 0001B         PUSHL   DESC+4                                      ;
00000000V  00            03  FB 0001E         CALLS   #3, NML_RECEIVE                             ;
           52            50  D0 00025         MOVL    R0, STATUS                                  ;
           0E            52  E9 00028         BLBC    STATUS, 2$                                  ; 0206
                         53  DD 0002B         PUSHL   R3                                          ; 0208
                    04   AE  9F 0002D         PUSHAB  DESC                                        ;
00000000G  00            02  FB 00030         CALLS   #2, NML$PROCESS_NICE                        ;
                         D3  11 00037         BRB     1$                                          ;
000020E4   8F            52  D1 00039 2$:     CMPL    STATUS, #8420                               ; 0218
                         28  13 00040         BEQL    4$                                          ;
000020EC   8F            52  D1 00042         CMPL    STATUS, #8428                               ; 0219
                         1F  13 00049         BEQL    4$                                          ;
00000838   8F            52  D1 0004B         CMPL    STATUS, #2104                               ; 0223
                         0B  12 00052         BNEQ    3$                                          ;
              00000000'  00  9F 00054         PUSHAB  P.AAB                                       ; 0225
           63            01  FB 0005A         CALLS   #1, NML_RESPONSE                            ;
                         AD  11 0005D         BRB     1$                                          ;
                         52  DD 0005F 3$:     PUSHL   STATUS                                      ; 0227
00000000G  00            01  FB 00061         CALLS   #1, LIB$STOP                                ;
                         A2  11 00068         BRB     1$                                          ; 0199
                             04 0006A 4$:     RET                                                 ; 0232

; Routine Size:  107 bytes,    Routine Base:  $CODE$ + 0024
;  237          0233  1 %SBTTL 'NML_INITIALIZE  Initialization routine'
;  238          0234  1 ROUTINE NML_INITIALIZE =
;  239          0235  1
;  240          0236  1 !++
;  241          0237  1 ! FUNCTIONAL DESCRIPTION:
;  242          0238  1 !
;  243          0239  1 !       This is the initialization routine for the DECnet-VAX Network
;  244          0240  1 !       Management Listener.  This module establishes (accepts) the logical
;  245          0241  1 !       link connection to NCP and initializes the data bases.
;  246          0242  1 !
;  247          0243  1 ! ROUTINE VALUE:
;  248          0244  1 ! COMPLETION CODES:
;  249          0245  1 !
;  250          0246  1 !       System service error is returned if initial assign for command link
;  251          0247  1 !       fails.  Subsequent system service failures cause a resource error
;  252          0248  1 !       (NML$_STS_RES) to be returned.  If the connect data is invalid or
;  253          0249  1 !       incompatible a compatibility error (NML$_STS_MVE) is retu-ned.
;  254          0250  1 !       If the connect accept fails a network error code is returned.
;  255          0251  1 !
;  256          0252  1 !--
;  257          0253  1
;  258          0254  2     BEGIN
;  259          0255  2
;  260          0256  2     LOCAL
;  261          0257  2         INI_IOSB : $IOSB,               ! Connect/reject I/O status block
;  262          0258  2         NCB_DSC  : DESCRIPTOR,          ! Connect NCB descriptor
;  263          0259  2         STATUS;                         ! Temporary status
;  264          0260  2 !
;  265          0261  2 ! Determine the mode in which NML is running.
;  266          0262  2 !
;  267          0263  2     NML_GETMODE ();
;  268          0264  2 !
;  269          0265  2 ! Dispatch to proper initialization routine.
;  270          0266  2 !
;  271          0267  2     SELECTU .NML$B_MODE OF
;  272          0268  2         SET
;  273          0269  2         [NML$_CONNECT]:
;  274          0270  2
;  275          0271  2             STATUS = NML_CONNECT ();
;  276          0272  2
;  277          0273  2         [OTHERWISE]:
;  278          0274  2
;  279          0275  2             STATUS = NML$_STS_MPR;
;  280          0276  2
;  281          0277  2         TES;
;  282          0278  2
;  283          0279  2     RETURN .STATUS                      ! Return status code
;  284          0280  2
;  285          0281  1     END;                                ! End of NML_INITIALIZE





                                                       000C 00000 NML_INITIALIZE:
                                                                           .WORD   Save R2,R3                                   ; 0234
                                     5E           10  C2 00002            SUBL2   #16, SP                                      ;
```

```
                     00000000V  00              00  FB 00005         CALLS   #0, NML_GETMODE                            ; 0263
                                53 00000000'    00  9A 0000C         MOVZBL  NML$B_MODE, R3                             ; 0267
                                52              01  D0 00013         MOVL    #1, R2                                     ;
                                01              53  91 00016         CMPB    R3, #1                                     ; 0269
                                                09  12 00019         BNEQ    1$                                         ;
                                                52  D4 0001B         CLRL    R2                                         ;
                     00000000V  00              00  FB 0001D         CALLS   #0, NML_CONNECT                            ; 0271
                                03              52  E9 00024 1$:     BLBC    R2, 2$                                     ; 0273
                                50              0A  CE 00027         MNEGL   #10, STATUS                                ; 0275
                                                    04 0002A 2$:     RET                                                ; 0281

; Routine Size:  43 bytes,    Routine Base:  $CODE$ + 008F

;  286          0282  1
```

```
  1 %SBTTL 'NML_CONNECT  Connect routine'
  1 ROUTINE NML_CONNECT =
  1
  1 !++
  1 ! FUNCTIONAL DESCRIPTION:
  1 !
  1 !       This establishes (accepts) the logical link connection to the
  1 !       command process.
  1 !
  1 ! IMPLICIT OUTPUTS:
  1 !       NML$W_CMD_CHAN  Channel number for logical link to NCP
  1 !
  1 ! ROUTINE VALUE:
  1 ! COMPLETION CODES:
  1 !
  1 !       System service error is returned if initial assign for command link
  1 !       fails.  Subsequent system service failures cause a resource error
  1 !       (NML$_STS_RES) tc be returned.  If the connect data is invalid or
  1 !       incompatible a compatibility error (NML$_STS_MVE) is returned.
  1 !       If the connect accept fails a network error code is returned.
  1 !
  1 !--
  1
  2 BEGIN
  2
  2 LOCAL
  2     INI_IOSB : $IOSB,           ! Connect/reject I/O status block
  2     NCBDSC   : DESCRIPTOR,
  2     STATUS;                             ! Temporary status
  2 !
  2 ! Assign channel for link to command process
  2 !
P 2 STATUS = $ASSIGN (DEVNAM = %ASCID '_NET:',
  2                   CHAN   = NML$W_CMD_CHAN);
  2 IF .STATUS THEN
  3     BEGIN
  3     !
  3     ! Check optional connect data for compatibility
  3     !
  3     STATUS = NML_VALIDLINK ();
  3     NCBDSC [DSC$A_POINTER] = .NML$A_NCBADDRESS;
  3     IF .STATUS THEN
  4         BEGIN
  4         !
  4         ! Set up NCB to accept the connection.  Add three bytes of version
  4         ! number to the NCB that was received.  Also use the length of the
  4         ! received NCB.
  4         !
  4         CH$WCHAR_A (3, NML$A_NCBDATA);
  4         IF CH$EQL (3, UPLIT BYTE (2,0,0),
  4                    3, .NML$A_NCBDATA) THEN
  4             CH$MOVE (3, UPLIT BYTE (2,0,0), .NML$A_NCBDATA)
  4         ELSE
  4             CH$MOVE (3, NML_VERSION, .NML$A_NCBDATA);
  4         NCBDSC [DSC$W_LENGTH] = .NML$L_CMDNCBLEN;
  4         !
  4         ! Accept connection to command process.
```

```
;  345          0340  4          !
;  346        P 0341  4          STATUS = $QIOW (CHAN = .NML$W_CMD_CHAN,
;  347        P 0342  4                          FUNC = IO$_ACCESS,
;  348        P 0343  4                          IOSB = INI_IOSB,
;  349          0344  4                          P2   = NCBDSC);
;  350          0345  4          IF .STATUS THEN
;  351          0346  4              STATUS = .INI_IOSB [IOS$W_STATUS]; ! Network error
;  352          0347  4
;  353          0348  4          END
;  354          0349  3      ELSE
;  355          0350  4          BEGIN
;  356          0351  4          !
;  357          0352  4          ! Reject connection due to incompatibility.  Add one byte reject code
;  358          0353  4          ! to the received NCB.
;  359          0354  4          !
;  360          0355  4          CH$WCHAR_A (1, NML$A_NCBDATA);
;  361          0356  4          CH$WCHAR (.STATUS / 2, .NML$A_NCBDATA);
;  362          0357  4          NCBDSC [DSC$W_LENGTH] = .NML$C_CMDNCBLEN;
;  363          0358  4
;  364        P 0359  4          $QIOW (CHAN = .NML$W_CMD_CHAN,             ! Channel
;  365        P 0360  4                  FUNC = IO$_ACCESS OR IO$M_ABORT,! Reject function code
;  366          0361  4                  P2   = NCBDSC);                    ! Reject NCB
;  367          0362  4
;  368          0363  3          END;
;  369          0364  2      END;
;  370          0365  2
;  371          0366  2 RETURN .STATUS                      ! Return status code
;  372          0367  2
;  373          0368  1 END;                                ! End of NML_CONNECT


                                                                   .PSECT  $PLIT$,NOWRT,NOEXE,2

                      00  00  00  3A  54  45  4E  5F  0000C P.AAE:  .ASCII  \_NET:\<0><0><0>                          ;
                                          010E0005  00014 P.AAD:  .LONG   17694725                                  ;
                                         00000000'  00018         .ADDRESS P.AAE                                    ;
                                        00  00  02  0001C P.AAF:  .BYTE   2, 0, 0                                   ;
                                        00  00  02  0001F P.AAG:  .BYTE   2, 0, 0                                   ;

                                                                  .EXTRN  SYS$ASSIGN, SYS$QIOW

                                                                  .PSECT  $CODE$,NOWRT,2

                                            07FC 00000 NML_CONNECT:
                                                                  .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10        ; 0284
                              5A 00000000G  00  9E 00002          MOVAB   SYS$QIOW, R10                             ;
                              59 00000000'  00  9E 00009          MOVAB   P.AAD, R9                                 ;
                              58 00000000'  00  9E 00010          MOVAB   NML$A_NCBDATA, R8                         ;
                              5E            10  C2 00017          SUBL2   #16, SP                                   ;
                                            7E  7C 0001A          CLRQ    -(SP)                                     ; 0316
                                     F8     A8  9F 0001C          PUSHAB  NML$W_CMD_CHAN                            ;
                                            59  DD 0001F          PUSHL   R9                                        ;
                      00000000G  00         04  FB 00021          CALLS   #4, SYS$ASSIGN                            ;
                                 56         50  D0 00028          MOVL    R0, STATUS                                ;
                                 5E         56  E9 0002B          BLBC    STATUS, 3$                                ; 0317
                      00000000V  00         00  FB 0002E          CALLS   #0, NML_VALIDLINK                         ; 0322
```

```
                                          56         50  D0 00035          MOVL    R0, STATUS                                        ;
                                    04    AE     FC  A8  D0 00038          MOVL    NML$A_NCBADDRESS, NCBDSC+4                        ; 0323
                                          50         68  D0 0003D          MOVL    NML$A_NCBDATA, R0                                 ; 0331
                                          57     F4  A8  D0 00040          MOVL    NML$L_CMDNCBLEN, R7                               ; 0337
                                          6E         57  B0 00044          MOVW    R7, NCBDSC                                        ;
                                          55     F8  A8  3C 00047          MOVZWL  NML$W_CMD_CHAN, R5                                ; 0344
                                          40         56  E9 0004B          BLBC    STATUS, 4$                                        ; 0324
                                          60         03  90 0004E          MOVB    #3, (R0)                                          ; 0331
                                                     68  D6 00051          INCL    NML$A_NCBDATA                                     ;
                                          54         68  D0 00053          MOVL    NML$A_NCBDATA, R4                                 ; 0333
                           64       08    A9         03  29 00056          CMPC3   #3, P.AAF, (R4)                                   ; 0332
                                                     08  12 0005B          BNEQ    1$                                                ;
          64               18             00     0B  A9  F0 0005D          INSV    P.AAG, #0, #24, (R4)                              ; 0334
                                                     06  11 00063          BRB     2$                                                ;
          64               18             00     EC  A9  F0 00065 1$:      INSV    NML_VERSION, #0, #24, (R4)                        ; 0336
                                                     7E  7C 0006B 2$:      CLRQ    -(SP)                                             ; 0344
                                                     7E  7C 0006D          CLRQ    -(SP)                                             ;
                                                 10  AE  9F 0006F          PUSHAB  NCBDSC                                            ;
                                                     7E  7C 00072          CLRQ    -(SP)                                             ;
                                                     7E  D4 00074          CLRL    -(SP)                                             ;
                                                 28  AE  9F 00076          PUSHAB  INI_IOSB                                          ;
                                                     32  DD 00079          PUSHL   #50                                               ;
                                                     55  DD 0007B          PUSHL   R5                                                ;
                                                     7E  D4 0007D          CLRL    -(SP)                                             ;
                                          6A         0C  FB 0007F          CALLS   #12, SYS$QIOW                                     ;
                                          56         50  D0 00082          MOVL    R0, STATUS                                        ;
                                          2C         56  E9 00085          BLBC    STATUS, 5$                                        ; 0345
                                          56     08  AE  3C 00088          MOVZWL  INI_IOSB, STATUS                                  ; 0346
                                                     26  11 0008C 3$:      BRB     5$                                                ; 0324
                                          60         01  90 0008E 4$:      MOVB    #1, (R0)                                          ; 0355
                                                     68  D6 00091          INCL    NML$A_NCBDATA                                     ;
                                          50         68  D0 00093          MOVL    NML$A_NCBDATA, R0                                 ; 0356
                           51             56         02  C7 00096          DIVL3   #2, STATUS, R1                                    ;
                                          60         51  90 0009A          MOVB    R1, (R0)                                          ;
                                                     7E  7C 0009D          CLRQ    -(SP)                                             ; 0361
                                                     7E  7C 0009F          CLRQ    -(SP)                                             ;
                                                 10  AE  9F 000A1          PUSHAB  NCBDSC                                            ;
                                                     7E  7C 000A4          CLRQ    -(SP)                                             ;
                                                     7E  7C 000A6          CLRQ    -(SP)                                             ;
                                          7E   0132  8F  3C 000A8          MOVZWL  #306, -(SP)                                       ;
                                                     55  DD 000AD          PUSHL   R5                                                ;
                                                     7E  D4 000AF          CLRL    -(SP)                                             ;
                                          6A         0C  FB 000B1          CALLS   #12, SYS$QIOW                                     ;
                                          50         56  D0 000B4 5$:      MOVL    STATUS, R0                                        ; 0366
                                                         04 000B7          RET                                                       ; 0368

; Routine Size:  184 bytes,    Routine Base:  $CODE$ + 00BA
```

```
1 %SBTTL 'NML_VALIDLINK  Version compatibilty check routine'
1 ROUTINE NML_VALIDLINK =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       The process attempting to connect to NML must pass an NCB which, in
1 !       the optional data field, contains a NICE version number.  This is
1 !       the version of NICE which the process expects to use in communicating
1 !       with NML.  This routine examines the version number to make sure it
1 !       is compatible with those versions of NICE spoken by this version of NML.
1 !
1 ! IMPLICIT INPUTS:
1 !       NML$A_NCBDATA contains the pointer to the NCB from NCP.
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NML$AB_RCVBUFFER contains the received NCB.
1 !       NML$L_CMDNCBLEN contains the length of the received NCB.
1 !       NML$A_NCBDATA contains the pointer to the optional NCB connect data
1 !               in NML$AB_RCVBUFFER.
1 !
1 ! ROUTINE VALUE:
1 ! COMPLETION CODES:
1 !
1 !       If receive Network Management version number is not greater
1 !       than or equal to our own then NML$_STS_MVE is returned otherwise
1 !       success (NML$_STS_SUC).
1 !
1 !--
1
2 BEGIN
2
2 LOCAL
2     NCP_VERSION : VECTOR [3,BYTE];      ! Command node NICE version number
2
2 !
2 ! Find optional data in NCB (two bytes past '/' character)
2 !
2 NML$A_NCBDATA = CH$FIND_CH (.NML$L_CMDNCBLEN,
2                             .NML$A_NCBADDRESS,
2                             %C'/');
2
2 IF NOT CH$FAIL (.NML$A_NCBDATA) THEN
2     NML$A_NCBDATA = .NML$A_NCBDATA + 3 ! Skip '/' and two bytes
2 ELSE
2     RETURN NML$_STS_MPR;        ! Return error
2 !
2 ! Compare received version with our own.  Value greater than or equal
2 ! is successful (Phase III).  Less than three bytes of optional data or
2 ! value less than our own but not version 2.0.0 or 3.0.0 fails.
2 !
2 IF CH$RCHAR (.NML$A_NCBDATA) EQLU 0 THEN
2     !
2     ! If no version number in the NCB, assume Phase II, NICE V1.3.0.
2     !
3     BEGIN
```

```
;  432          0426  3     CH$MOVE (3, UPLIT BYTE(1,3,0), NCP_VERSION);
;  433          0427  3     END
;  434          0428  2 ELSE
;  435          0429  3     BEGIN
;  436          0430  3     IF CH$RCHAR (.NML$A_NCBDATA) EQLU 3 THEN
;  437          0431  3         CH$MOVE (3, .NML$A_NCBDATA+1, NCP_VERSION)
;  438          0432  3     ELSE
;  439          0433  3         RETURN NML$_STS_MVE;
;  440          0434  2     END;
;  441          0435  2 IF NML$INITIALIZE (NCP_VERSION) THEN
;  442          0436  2     RETURN NML$_STS_SUC
;  443          0437  2 ELSE
;  444          0438  2     RETURN NML$_STS_MVE;
;  445          0439  1 END;                            ! End of NML_VALIDLINK



                                                                          .PSECT  $PLIT$,NOWRT,NOEXE,2

                                                00  03  01  00022 P.AAH:  .BYTE   1, 3, 0                                           ;



                                                                          .PSECT  $CODE$,NOWRT,2

                                                       0004 00000 NML_VALIDLINK:
                                                                          .WORD   Save R2                                           ; 0370
                                       52 00000000'  00  9E 00002         MOVAB   NML$A_NCBDATA, R2                                 ;
                                       5E            04  C2 00009         SUBL2   #4, SP                                            ;
                                       50       FC   A2  D0 0000C         MOVL    NML$A_NCBADDRESS, R0                              ; 0409
                         60       F4   A2            2F  3A 00010         LOCC    #47, NML$L_CMDNCBLEN, (R0)                        ; 0408
                                                     02  12 00015         BNEQ    1$                                                ;
                                                     51  D4 00017         CLRL    R1                                                ;
                                       62            51  D0 00019 1$:     MOVL    R1, NML$A_NCBDATA                                 ;
                                                     05  13 0001C         BEQL    2$                                                ; 0412
                                       62            03  C0 0001E         ADDL2   #3, NML$A_NCBDATA                                 ; 0413
                                                     04  11 00021         BRB     3$                                                ;
                                       50            0A  CE 00023 2$:     MNEGL   #10, R0                                           ; 0415
                                                         04 00026         RET                                                       ;
                                       50            62  D0 00027 3$:     MOVL    NML$A_NCBDATA, R0                                 ; 0421
                                                     60  95 0002A         TSTB    (R0)                                              ;
                                                     0B  12 0002C         BNEQ    4$                                                ;
           6E            18            00 00000000'  00  F0 0002E         INSV    P.AAH, #0, #24, NCP_VERSION                       ; 0426
                                                     0B  11 00037         BRB     5$                                                ; 0421
                                       03            60  91 00039 4$:     CMPB    (R0), #3                                          ; 0430
                                                     16  12 0003C         BNEQ    6$                                                ;
           6E            18            00       01   A0  F0 0003E         INSV    1(R0), #0, #24, NCP_VERSION                       ; 0431
                                                     5E  DD 00044 5$:     PUSHL   SP                                                ; 0435
                            00000000G  00            01  FB 00046         CALLS   #1, NML$INITIALIZE                                ;
                                       04            50  E9 0004D         BLBC    R0, 6$                                            ;
                                       50            01  D0 00050         MOVL    #1, R0                                            ; 0438
                                                         04 00053         RET                                                       ;
                                       50            0E  CE 00054 6$:     MNEGL   #14, R0                                           ;
                                                         04 00057         RET                                                       ; 0439

; Routine Size:  88 bytes,    Routine Base:  $CODE$ + 0172
```

```
:  447        0440   1 %SBTTL 'NML_GETMODE  Determine the mode in which NML is running'
:  448        0441   1 ROUTINE NML_GETMODE : NOVALUE =
:  449        0442   1
:  450        0443   1 !++
:  451        0444   1 ! FUNCTIONAL DESCRIPTION:
:  452        0445   1 !       This routine determines the mode in which NML is running by
:  453        0446   1 !       determining how the image was activated.  NML runs one of in two
:  454        0447   1 !       modes:
:  455        0448   1 !               Communicating with NCP on a remote node via a logical link,
:  456        0449   1 !                       processing NICE commands.
:  457        0450   1 !               In autoservice mode - doing down line loads and loopback
:  458        0451   1 !                       functions requested by other nodes using the MOP
:  459        0452   1 !                       protocol.
:  460        0453   1 !
:  461        0454   1 ! FORMAL PARAMETERS:
:  462        0455   1 !
:  463        0456   1 !       NONE
:  464        0457   1 !
:  465        0458   1 ! IMPLICIT INPUTS:
:  466        0459   1 !
:  467        0460   1 !       NONE
:  468        0461   1 !
:  469        0462   1 ! IMPLICIT OUTPUTS:
:  470        0463   1 !
:  471        0464   1 !       NML$B_MODE contains the mode in which NML is running.
:  472        0465   1 !               (it may be unchanged if we cannot determine the mode)
:  473        0466   1 !
:  474        0467   1 ! ROUTINE VALUE:
:  475        0468   1 ! COMPLETION CODES:
:  476        0469   1 !
:  477        0470   1 !       NONE
:  478        0471   1 !
:  479        0472   1 ! SIDE EFFECTS:
:  480        0473   1 !
:  481        0474   1 !       NONE
:  482        0475   1 !
:  483        0476   1 !--
:  484        0477   1
:  485        0478   2     BEGIN
:  486        0479   2
:  487        0480   2     LOCAL
:  488        0481   2         STATUS;                         ! Temporary status
:  489        0482   2
:  490        0483   2 !
:  491        0484   2 ! Get optional connect data or service circuit name by translating SYS$NET
:  492        0485   2 !
:  493      P 0486   2     STATUS = $TRNLNM (ATTR = UPLIT (LNM$M_CASE_BLIND),
:  494      P 0487   2                       TABNAM = %ASCID 'LNM$PROCESS_TABLE',
:  495      P 0488   2                       LOGNAM = %ASCID 'SYS$NET',
:  496      P 0489   2                       ITMLST = UPLIT (WORD (nml$k_rcvbflen),
:  497      P 0490   2                                       WORD (lnm$_string),
:  498      P 0491   2                                       LONG (nml$ab_rcvbuffer),
:  499      P 0492   2                                       LONG (nml$l_cmdncblen),
:  500      P 0493   2                                       LONG (0))
:  501        0494   2                       );
:  502        0495   2     IF .STATUS EQLU SS$_NORMAL
:  503        0496   2     THEN
```

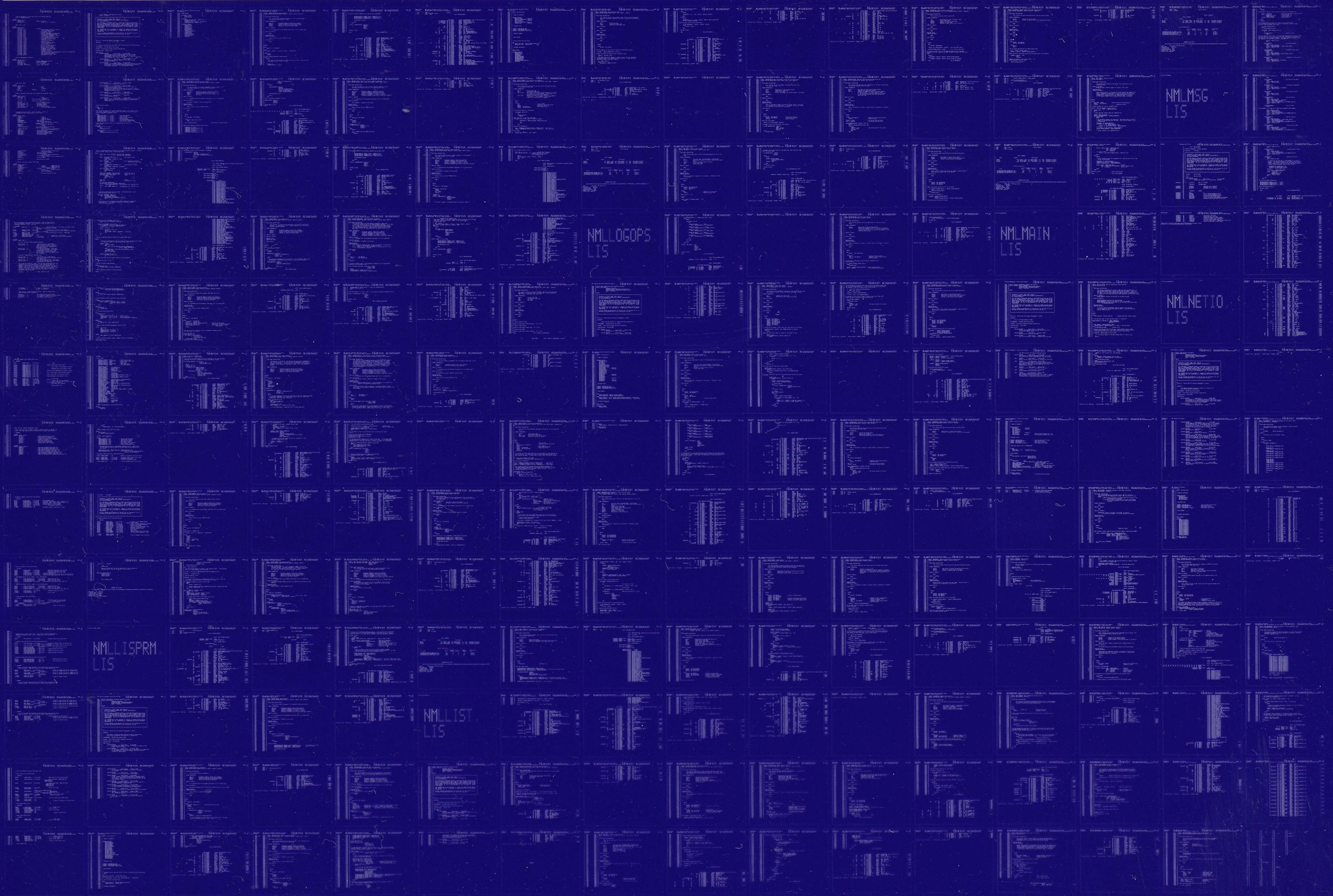
```
;  504          0497  3         BEGIN
;  505          0498  3         !
;  506          0499  3         ! Since SYS$NET translates, NML was activated by a connect renuest
;  507          0500  3         ! from an NCP on a remote node.
;  508          0501  3         !
;  509          0502  3         NML$B_MODE = NML$_CONNECT;
;  510          0503  3         NML$A_NCBADDRESS = NML$AB_RCVBUFFER; ! Initialize NCB pointer
;  511          0504  2         END;
;  512          0505  2
;  513          0506  1     END;                                ! End of NML_GETMODE


                                                                          .PSECT  $PLIT$,NOWRT,NOEXE,2

                                                            00025         .BLKB   3
                                                  02000000  00028 P.AAI:  .LONG   33554432
42  41  54  5F  53  53  45  43  4F  52  50  24  4D  4E  4C  0002C P.AAK:  .ASCII  \LNM$PROCESS_TABLE\<0><0><0>          ;
                                        00  00  00  45  4C  0003B                                                      ;
                                                  010E0011  00040 P.AAJ:  .LONG   17694737                              ;
                                                  00000000' 00044         .ADDRESS P.AAK                                ;
                            00  54  45  4E  24  53  59  53  00048 P.AAM:  .ASCII  \SYS$NET\<0>                          ;
                                                  010E0007  00050 P.AAL:  .LONG   17694727                              ;
                                                  00000000' 00054         .ADDRESS P.AAM                                ;
                                                      0200  00058 P.AAN:  .WORD   512                                   ;
                                                      0002  0005A         .WORD   2                                     ;
                                                  00000000' 0005C         .ADDRESS NML$AB_RCVBUFFER                     ;
                                                  00000000' 00060         .ADDRESS NML$L_CMDNCBLEN                      ;
                                                  00000000  00064         .LONG   0                                     ;

                                                                          .EXTRN  SYS$TRNLNM

                                                                          .PSECT  $CODE$,NOWRT,2

                                                       000C 00000 NML_GETMODE:
                                                                          .WORD   Save R2,R3                            ; 0441
                                       53 00000000'  00  9E 00002         MOVAB   NML$B_MODE, R3                        ;
                                       52 00000000'  00  9E 00009         MOVAB   P.AAN, R2                             ;
                                                     52  DD 00010         PUSHL   R2                                    ; 0494
                                                     7E  D4 00012         CLRL    -(SP)                                 ;
                                                F8   A2  9F 00014         PUSHAB  P.AAL                                 ;
                                                E8   A2  9F 00017         PUSHAB  P.AAJ                                 ;
                                                D0   A2  9F 0001A         PUSHAB  P.AAI                                 ;
                            00000000G  00            05  FB 0001D         CALLS   #5, SYS$TRNLNM                        ;
                                       01            50  D1 00024         CMPL    STATUS, #1                            ; 0495
                                                     08  12 00027         BNEQ    1$                                    ;
                                       63            01  90 00029         MOVB    #1, NML$B_MODE                        ; 0502
                                  0C   A3       1C   A3  9E 0002C         MOVAB   NML$AB_RCVBUFFER, NML$A_NCBADDRESS    ; 0503
                                                         04 00031 1$:     RET                                           ; 0506

; Routine Size:  50 bytes,    Routine Base:  $CODE$ + 01CA
;   515          0507   1 %SBTTL 'NML_RECEIVE  Network receive routine'
;   516          0508   1 ROUTINE NML_RECEIVE (BUFADR, BUFLEN, RCVLEN) =
;   517          0509   1
;   518          0510   1 !++
;   519          0511   1 ! FUNCTIONAL DESCRIPTION:
;   520          0512   1 !
;   521          0513   1 !       This routine receives NICE protocol messages over a logical link
;   522          0514   1 !       from a command process (generally NCP).
;   523          0515   1 !
;   524          0516   1 ! FORMAL PARAMETERS:
;   525          0517   1 !
;   526          0518   1 !       BUFADR          Address of the receive buffer.
;   527          0519   1 !       BUFLEN          Length of the receive buffer.
;   528          0520   1 !       RCVLEN          Actual length of the received data if successful.
;   529          0521   1 !
;   530          0522   1 ! IMPLICIT INPUTS:
;   531          0523   1 !
;   532          0524   1 !       NML$W_CMD_CHAN  Channel assigred to the command process link.
;   533          0525   1 !
;   534          0526   1 ! IMPLICIT OUTPUTS:
;   535          0527   1 !
;   536          0528   1 !       The buffer pointed to by BUFADR contains the received data.
;   537          0529   1 !
;   538          0530   1 ! ROUTINE VALUE:
;   539          0531   1 ! COMPLETION CODES:
;   540          0532   1 !
;   541          0533   1 !       System service completion status is returned if receive fails
;   542          0534   1 !       indicating that the link has been terminated for some reason.
;   543          0535   1 !       Otherwise success is returned.
;   544          0536   1 !
;   545          0537   1 ! SIDE EFFECTS:
;   546          0538   1 !
;   547          0539   1 !       NONE
;   548          0540   1 !
;   549          0541   1 !--
;   550          0542   1
;   551          0543   2     BEGIN
;   552          0544   2
;   553          0545   2     LOCAL
;   554          0546   2         RCV_IOSB : $IOSB,                         ! Receive I/O status block
;   555          0547   2         STATUS;                                   ! Temporary status
;   556          0548   2 !
;   557          0549   2 ! Receive a NICE message
;   558          0550   2 !
;   559        P 0551   2     STATUS = $QIOW (FUNC = IO$_READVBLK,          ! Function
;   560        P 0552   2                     CHAN = .NML$W_CMD_CHAN,       ! Channel
;   561        P 0553   2                     IOSB = RCV_IOSB,              ! I/O status block
;   562        P 0554   2                     P1   = .BUFADR,               ! Buffer address
;   563          0555   2                     P2   = .BUFLEN);              ! Buffer length
;   564          0556   2 !
;   565          0557   2 ! If system service completed successfully get network status
;   566          0558   2 !
;   567          0559   2     IF .STATUS
;   568          0560   2     THEN
;   569          0561   2         STATUS = .RCV_IOSB [IOS$W_STATUS];
;   570          0562   2 !
;   571          0563   2 ! If network status is success then get received message length
;  572          0564  2 !
;  573          0565  2        IF .STATUS
;  574          0566  2        THEN
;  575          0567  2            .RCVLEN = .RCV_IOSB [IOS$W_COUNT];        ! Return received data length
;  576          0568  2
;  577          0569  2        RETURN .STATUS                                 ! Return status code
;  578          0570  2
;  579          0571  1        END;                                           ! End of NML$RECEIVE



                                            0000 00000 NML_RECEIVE:
                                                              .WORD   Save nothing                                      ; 0508
                              5E         08  C2 00002         SUBL2   #8, SP                                            ;
                                         7E  7C 00005         CLRQ    -(SP)                                             ; 0555
                                         7E  7C 00007         CLRQ    -(SP)                                             ;
                              7E      04  AC  7D 00009         MOVQ    BUFADR, -(SP)                                     ;
                                         7E  7C 0000D         CLRQ    -(SP)                                             ;
                                      20  AE  9F 0000F         PUSHAB  RCV_IOSB                                          ;
                                         31  DD 00012         PUSHL   #49                                               ;
                              7E 00000000'  00  3C 00014         MOVZWL  NML$W_CMD_CHAN, -(SP)                             ;
                                         7E  D4 0001B         CLRL    -(SP)                                             ;
                    00000000G  00         0C  FB 0001D         CALLS   #12, SYS$QIOW                                     ;
                               0B         50  E9 00024         BLBC    STATUS, 1$                                        ; 0559
                               50         6E  3C 00027         MOVZWL  RCV_IOSB, STATUS                                  ; 0561
                               05         50  E9 0002A         BLBC    STATUS, 1$                                        ; 0565
                         0C    BC     02  AE  3C 0002D         MOVZWL  RCV_IOSB+2, @RCVLEN                               ; 0567
                                         04 00032 1$:         RET                                                       ; 0571

; Routine Size:  51 bytes,    Routine Base:  $CODE$ + 01FC


;  580          0572  1
```

```
  1 %SBTTL 'NML_RESPONSE Network response routine'
  1 ROUTINE NML_RESPONSE (BUFDESC): NOVALUE =
  1
  1 !++
  1 !
  1 !       This routine is called to process a NICE response as a result
  1 !       of processing a NICE message.  This routine may be called as
  1 !       many times as necessary during the processing of a single NICE
  1 !       message.
  1 !
  1 ! Inputs:
  1 !
  1 !       bufdesc         Address of descriptor of message to be transmitted.
  1 !
  1 !       nml$w_cmd_chan  Channel assigned to the command process link.
  1 !
  1 ! Outputs:
  1 !
  1 !       Returns success.  Errors are signalled.
  1 !--
  1
  2 BEGIN
  2
  2 MAP
  2     bufdesc:    REF BLOCK [,BYTE];      ! Address of descriptor
  2
  2 LOCAL
  2     snd_iosb : $IOSB,                   ! Send I/O status block
  2     status;                             ! Temporary status
  2
  2 !
  2 ! Send the NICE response message
  2 !
  2
P 2 status = $QIOW (FUNC = IO$_WRITEVBLK,
P 2                     CHAN = .nml$w_cmd_chan,
P 2                     IOSB = snd_iosb,
P 2                     P1   = .bufdesc [dsc$a_pointer],
  2                     P2   = .bufdesc [dsc$w_length]);
  2 !
  2 ! If system service was successful then get network status
  2 !
  2
  2 IF .STATUS
  2 THEN
  2     STATUS = .SND_IOSB [IOS$W_STATUS];
  2
  2 !
  2 ! If status is bad then signal the error.
  2 !
  2
  2 IF NOT .STATUS
  2 THEN
  2     SIGNAL_STOP (.STATUS);
  2
  1 END;
```

```
                                     0000 00000 NML_RESPONSE:
                                                        .WORD   Save nothing                            ; 0574
                  5E          08  C2 00002              SUBL2   #8, SP                                  ;
                              7E  7C 00005              CLRQ    -(SP)                                   ; 0611
                              7E  7C 00007              CLRQ    -(SP)                                   ;
                  50     04   AC  D0 00009              MOVL    BUFDESC, R0                             ;
                  7E          60  3C 0000D              MOVZWL  (R0), -(SP)                             ;
                         04   A0  DD 00010              PUSHL   4(R0)                                   ;
                              7E  7C 00013              CLRQ    -(SP)                                   ;
                         20   AE  9F 00015              PUSHAB  SND_IOSB                                ;
                              30  DD 00018              PUSHL   #48                                     ;
                  7E 00000000'  00  3C 0001A            MOVZWL  NML$W_CMD_CHAN, -(SP)                   ;
                              7E  D4 00021              CLRL    -(SP)                                   ;
        00000000G 00          0C  FB 00023              CALLS   #12, SYS$QIOW                           ;
                  06          50  E9 0002A              BLBC    STATUS, 1$                              ; 0616
                  50          6E  3C 0002D              MOVZWL  SND_IOSB, STATUS                        ; 0618
                  09          50  E8 00030              BLBS    STATUS, 2$                              ; 0624
                              50  DD 00033 1$:          PUSHL   STATUS                                  ; 0626
        00000000G 00          01  FB 00035              CALLS   #1, LIB$STOP                            ;
                                  04 0003C 2$:          RET                                             ; 0628

; Routine Size:  61 bytes,    Routine Base:  $CODE$ + 022F
;  639          0629  1 END                                   ! End of module
;  640          0630  0 ELUDOM



                                                                  .EXTRN  LIB$STOP

;                                 PSECT SUMMARY

;       Name                    Bytes                       Attributes

;   $PLIT$                          104  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $OWN$                           540  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $CODE$                          620  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                          Library Statistics
;
;                                            -------- Symbols --------      Pages       Processing
;       File                                 Total    Loaded   Percent      Mapped      Time
;
;   _$255$DUA28:[NML.OBJ]NMLLIB.L32;1         341        15         4         27         00:00.1
;   _$255$DUA28:[SHRLIB]NMALIBRY.L32;1        887         1         0         47         00:00.2
;   _$255$DUA28:[SHRLIB]NET.L32;1            1279         0         0         63         00:00.3
;   _$255$DUA28:[SYSLIB]STARLET.L32;1        9776        18         0        581         00:03.3




;                                 COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:NMLMAIN/OBJ=OBJ$:NMLMAIN MSRC$:NMLMAIN/UPDATE=(ENH$:NMLMAIN)

; Size:          620 code + 644 data bytes
; Run Time:         00:16.8
; Elapsed Time:     00:41.9
; Lines/CPU Min:     2255
; Lexemes/CPU-Min: 14233
; Memory Used:  101 pages
; Compilation Complete
```

0284 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



NMLMSG
LIS

NMLLOGOPS
LIS

NMLMAIN
LIS

NMLNETIO
LIS

NMLLISPRM
LIS

NMLLIST
LIS