```
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
```

FILEID**NMLUTIL

```
NN      NN  MM      MM  LL          UU      UU  TTTTTTTTTT    IIIIII    LL
NN      NN  MM      MM  LL          UU      UU  TTTTTTTTTT    IIIIII    LL
NN      NN  MMMM  MMMM  LL          UU      UU      TT          II      LL
NN      NN  MMMM  MMMM  LL          UU      UU      TT          II      LL
NNNN    NN  MM  MM  MM  LL          UU      UU      TT          II      LL
NNNN    NN  MM  MM  MM  LL          UU      UU      TT          II      LL
NN  NN  NN  MM      MM  LL          UU      UU      TT          II      LL
NN  NN  NN  MM      MM  LL          UU      UU      TT          II      LL
NN    NNNN  MM      MM  LL          UU      UU      TT          II      LL
NN    NNNN  MM      MM  LL          UU      UU      TT          II      LL
NN      NN  MM      MM  LL          UU      UU      TT          II      LL
NN      NN  MM      MM  LL          UU      UU      TT          II      LL
NN      NN  MM      MM  LLLLLLLLLL  UUUUUUUUUU      TT      IIIIII    LLLLLLLLLL
NN      NN  MM      MM  LLLLLLLLLL  UUUUUUUUUU      TT      IIIIII    LLLLLLLLLL

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
```

```
;    1          0001  0 %TITLE 'NML Utility routines'
;    2          0002  0 MODULE NML$UTILITY (
;    3          0003  0                     LANGUAGE (BLISS32),
;    4          0004  0                     ADDRESSING_MODE (NONEXTERNAL=GENERAL),
;    5          0005  0                     ADDRESSING_MODE (EXTERNAL=GENERAL),
;    6          0006  0                     IDENT = 'V04-000'
;    7          0007  0                     ) =
;    8          0008  1 BEGIN
;    9          0009  1 !
;   10          0010  1 !*****************************************************************************
;   11          0011  1 !*                                                                           *
;   12          0012  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;   13          0013  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;   14          0014  1 !*  ALL RIGHTS RESERVED.                                                      *
;   15          0015  1 !*                                                                           *
;   16          0016  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;   17          0017  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;   18          0018  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;   19          0019  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;   20          0020  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;   21          0021  1 !*  TRANSFERRED.                                                              *
;   22          0022  1 !*                                                                           *
;   23          0023  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;   24          0024  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;   25          0025  1 !*  CORPORATION.                                                              *
;   26          0026  1 !*                                                                           *
;   27          0027  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;   28          0028  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;   29          0029  1 !*                                                                           *
;   30          0030  1 !*                                                                           *
;   31          0031  1 !*****************************************************************************
;   32          0032  1 !
;   33          0033  1
;   34          0034  1 !++
;   35          0035  1 ! FACILITY:  DECnet-VAX V2.0 Network Management Listener
;   36          0036  1 !
;   37          0037  1 ! ABSTRACT:
;   38          0038  1 !
;   39          0039  1 !       This module contains routines for handling a variety of common
;   40          0040  1 !       functions.
;   41          0041  1 !
;   42          0042  1 ! ENVIRONMENT:  VAX/VMS Operating System
;   43          0043  1 !
;   44          0044  1 ! AUTHOR:  Distributed Systems Software Engineering
;   45          0045  1 !
;   46          0046  1 ! CREATION DATE:  23-JAN-1980
;   47          0047  1 !
;   48          0048  1 ! MODIFIED BY:
;   49          0049  1 !
;   50          0050  1 !       V03-009 MKP0009         Kathy Perko             23-July-1984
;   51          0051  1 !               Change area number defaulting so that, if no area number
;   52          0052  1 !               is supplied in the NICE command, the executor node's area
;   53          0053  1 !               is used.  This means the permanent database executor area
;   54          0054  1 !               number for permanent database operations, and the volatile
;   55          0055  1 !               database executor area for volatile database operations.
;   56          0056  1 !
;   57          0057  1 !       V03-008 MKP0008         Kathy Perko             21-June-1984
```

```
;    58          0058  1 !                 Modify NML$CHKEXE to return success if the node address
;    59          0059  1 !                 being checked is 0.
;    60          0060  1 !
;    61          0061  1 !       V03-007 MKP0007         Kathy Perko             19-April-1984
;    62          0062  1 !                 Modify NML$GETEXEID to call NML$GETEXENAM instead of
;    63          0063  1 !                 NML$GETNODNAM.
;    64          0064  1 !
;    65          0065  1 !       V03-006 MKP0006         Kathy Perko             18-April-1984
;    66          0066  1 !                 Fix NML$CHKEXEID so it's checking only a word for the
;    67          0067  1 !                 node address (instead of a longword).
;    68          0068  1 !
;    69          0069  1 !       V03-005 MKP0005         Kathy Perko             25-Mar-1984
;    70          0070  1 !                 Add a routine to check a node number, and, if it's got
;    71          0071  1 !                 an area = 0, then convert it to 1 if talking to a Phase IV
;    72          0072  1 !                 NCP, and convert it to the exec's area if talking to a
;    73          0073  1 !                 Phase III NCP.
;    74          0074  1 !                 Use global executor node addresses.
;    75          0075  1 !
;    76          0076  1 !       V03-004 MKP0004         Kathy Perko             5-Feb-1984
;    77          0077  1 !                 Make sure permanent database file opens are done at the
;    78          0078  1 !                 right times.
;    79          0079  1 !
;    80          0080  1 !       V03-003 MKP0003         Kathy Perko             4-Aug-1983
;    81          0081  1 !                 Make changes to convert node permanent database to utilize
;    82          0082  1 !                 multiple ISAM keys.  This should improve performance.
;    83          0083  1 !
;    84          0084  1 !       V03-002 MKP0002         Kathy Perko             21-June-1982
;    85          0085  1 !                 Add to NML$BLDP2 so that it will take search key values
;    86          0086  1 !                 with a word length.
;    87          0087  1 !
;    88          0088  1 !       V03-001 MKP0001         Kathy Perko             21-April-1982
;    89          0089  1 !                 Change NML$BLDP2 to build P2 buffers with second start key
;    90          0090  1 !                 and no start key.  Also, always include a context area.
;    91          0091  1 !                 Add support for entity qualifiers.
;    92          0092  1 !
;    93          0093  1 !       V02-001 LMK0001         Len Kawell              21-Jul-1981
;    94          0094  1 !                 Modifications for new NETACP control QIO.
;    95          0095  1 !--
;    96          0096  1
```

```
;    98          0097  1 %SBTTL  'Declarations'
;    99          0098  1
;   100          0099  1 !
;   101          0100  1 ! TABLE OF CONTENTS:
;   102          0101  1 !
;   103          0102  1
;   104          0103  1 FORWARD ROUTINE
;   105          0104  1     NML$BLDP2               : NOVALUE,
;   106          0105  1     NML$CHKEXE,
;   107          0106  1     NML$SET_UP_EXEC_ID,
;   108          0107  1     NML$GETEXEADR,
;   109          0108  1     NML$GETEXENAM,
;   110          0109  1     NML$GETNODNAM,
;   111          0110  1     NML$GETVOLNDNAM,
;   112          0111  1     NML$GETNODADR,
;   113          0112  1     NML$GETVOLNDADR,
;   114          0113  1     NML$GETEXEID,
;   115          0114  1     NML$GETINFTABS,
;   116          0115  1     NML$FIX_NODE_NUM;
;   117          0116  1
;   118          0117  1 !
;   119          0118  1 ! INCLUDE FILES:
;   120          0119  1 !
;   121          0120  1
;   122          0121  1 LIBRARY 'LIB$:NMLLIB.L32';
;   123          0122  1 LIBRARY 'SHRLIB$:NMALIBRY.L32';
;   124          0123  1 LIBRARY 'SHRLIB$:NET.L32';
;   125          0124  1 LIBRARY 'SYS$LIBRARY:STARLET.L32';
;   126          0125  1
;   127          0126  1 !
;   128          0127  1 ! OWN STORAGE:
;   129          0128  1 !
;   130          0129  1 !
;   131          0130  1 ! Many NICE commands need the executor node's address and/or name.  Save them
;   132          0131  1 ! here.  The volatile database exec name and address can't change when
;   133          0132  1 ! the exec's state is ON, so they are only retrieved once for each run of
;   134          0133  1 ! NMLSHR.  The permanent database exec name and address are retrieved, at
;   135          0134  1 ! most, once per NICE command.  They are not retrieved if they are not needed.
;   136          0135  1 !
;   137          0136  1 GLOBAL
;   138          0137  1     nml$gw_vol_exec_addr:       WORD,
;   139          0138  1     nml$gw_perm_exec_addr:      WORD,
;   140          0139  1     nml$t_vol_exec_name:        BBLOCK [16],
;   141          0140  1     nml$gq_vol_exec_name_dsc:   VECTOR [2] INITIAL (0, nml$t_vol_exec_name),
;   142          0141  1     nml$t_perm_exec_name:       BBLOCK [16],
;   143          0142  1     nml$gq_perm_exec_name_dsc:  VECTOR [2] INITIAL (0, nml$t_perm_exec_name);
;   144          0143  1
;   145          0144  1 !
;   146          0145  1 ! Parameter buffers and descriptors for use in handling volatile data base
;   147          0146  1 ! data.
;   148          0147  1 !
;   149          0148  1 OWN
;   150          0149  1     p2buffer   : VECTOR [nml$k_p2buflen,  BYTE],
;   151          0150  1     prmbuffer : VECTOR [256, BYTE];
;   152          0151  1
;   153          0152  1 BIND
;   154          0153  1     p2bfdsc   = UPLIT (nml$k_p2buflen, p2buffer) : VECTOR [2],
```

```
;  155          0154  1      prmdsc = UPLIT (256, prmbuffer) : VECTOR [2];
;  156          0155  1
;  157          0156  1  OWN
;  158          0157  1      msglength;
;  159          0158  1
;  160          0159  1  !
;  161          0160  1  ! EXTERNAL REFERENCES:
;  162          0161  1  !
;  163          0162  1
;  164          0163  1  $nml_extdef;
;  165          0164  1
;  166          0165  1  EXTERNAL
;  167          0166  1      nml$gb_ncp_version;
;  168          0167  1
;  169          0168  1  EXTERNAL LITERAL
;  170          0169  1      nml$_qiobfovf;
;  171          0170  1
;  172          0171  1  EXTERNAL ROUTINE
;  173          0172  1      nml$bld_reply,
;  174          0173  1      nml$openfile,
;  175          0174  1      nma$searchfld,
;  176          0175  1      nml$error_1,
;  177          0176  1      nml$readrecord,
;  178          0177  1      nml$netqio;
;  179          0178  1
```

```
;  181     0179  1 %SBTTL 'NML$BLDP2  Build P2 buffer and descriptor'
;  182     0180  1 GLOBAL ROUTINE NML$BLDP2 (LEN1, ADR1, LEN2, ADR2, P2DSC, RESDSC) : NOVALUE =
;  183     0181  1
;  184     0182  1 !++
;  185     0183  1 ! FUNCTIONAL DESCRIPTION:
;  186     0184  1 !
;  187     0185  1 !       This routine builds the P2 buffer and descriptor for show operations.
;  188     0186  1 !       The search key is added followed by the start key.
;  189     0187  1 !
;  190     0188  1 ! FORMAL PARAMETERS:
;  191     0189  1 !
;  192     0190  1 !       LEN1            First search key length.  If LEN1 is:
;  193     0191  1 !                           - zero then ADR1 contains a longword search key.
;  194     0192  1 !                           - >0 it contains the length of a string which
;  195     0193  1 !                             ADR1 points to.
;  196     0194  1 !                           - -1 then search key ID is a wildcard, and nothing
;  197     0195  1 !                             needs to be put into the P2 buffer for it.
;  198     0196  1 !                           - -2 then ADR1 contains a word search key.
;  199     0197  1 !       ADR1            First search key address.  If LEN1 is zero then this
;  200     0198  1 !                       is the longword value of the search key.  If LEN1 is -1 then
;  201     0199  1 !                       the search key is omitted.
;  202     0200  1 !       LEN2            Second search  key length.  Same rules apply as for
;  203     0201  1 !                       LEN1.
;  204     0202  1 !       ADR2            Second search key address. Same rules apply as for
;  205     0203  1 !                       ADR1.
;  206     0204  1 !       P2DSC           Address of P2 descriptor.  This routine assumes that
;  207     0205  1 !                       the buffer is largest enough to handle the result.
;  208     0206  1 !                       The maximum P2 buffer required by NML is 36 bytes.
;  209     0207  1 !       RESDSC          Address of descriptor to hold resulting P2.
;  210     0208  1 !
;  211     0209  1 ! IMPLICIT OUTPUTS:
;  212     0210  1 !       The buffer described by P2DSC contains the search key and
;  213     0211  1 !       start key information.
;  214     0212  1 !
;  215     0213  1 !--
;  216     0214  1
;  217     0215  2 BEGIN
;  218     0216  2
;  219     0217  2 MAP
;  220     0218  2     P2DSC  : REF DESCRIPTOR,
;  221     0219  2     RESDSC : REF DESCRIPTOR;
;  222     0220  2
;  223     0221  2 OWN
;  224     0222  2     COLLATE_START_VALUE: VECTOR [NFB$C_CTX_SIZE, BYTE]
;  225     0223  2                          INITIAL ( REP NFB$C_CTX_SIZE OF BYTE (0));
;  226     0224  2
;  227     0225  2 LOCAL
;  228     0226  2     MSGSIZE,
;  229     0227  2     COUNT,                              ! P2 buffer length
;  230     0228  2     PTR;                                ! P2 buffer pointer
;  231     0229  2
;  232     0230  2 !
;  233     0231  2 ! Calculate the length of the resulting P2 buffer, and signal if
;  234     0232  2 ! the buffer supplied isn't big enough.
;  235     0233  2 !
;  236     0234  2 COUNT = 4;                          ! Account for count at beginning of buffer.
;  237     0235  2 SELECTONE .LEN1 OF
```

```
;  238     0236  2      SET
;  239     0237  2      [-2]: COUNT = .COUNT + 2;                       ! It's a word
;  240     0238  2      [0]: COUNT = .COUNT + 4;                        ! It's a longword
;  241     0239  2      [1 TO 255]: COUNT = .COUNT + .LEN1 + 2          ! It's a string.
;  242     0240  2      TES;
;  243     0241  2
;  244     0242  2  SELECTONE .LEN2 OF
;  245     0243  2      SET
;  246     0244  2      [-2]: COUNT = .COUNT + 2;                       ! It's a word
;  247     0245  2      [0]: COUNT = .COUNT + 4;                        ! It's a longword
;  248     0246  2      [1 TO 255]: COUNT = .COUNT + .LEN2 + 2          ! It's a string.
;  249     0247  2      TES;
;  250     0248  2
;  251     0249  2  COUNT = .COUNT + NFB$C_CTX_SIZE;
;  252     0250  2  IF .COUNT GTR .P2DSC [DSC$W_LENGTH] THEN
;  253     0251  2      !
;  254     0252  2      ! The P2 buffer will overflow.  Signal an NML error.
;  255     0253  2      !
;  256     0254  3      BEGIN
;  257     0255  3      NML$AB_MSGBLOCK [MSB$L_FLAGS] = MSB$M_MSG_FLD; ! Set message text flag.
;  258     0256  3      NML$AB_MSGBLOCK [MSB$B_CODE] = NMA$C_STS_MPR;
;  259     0257  3      NML$AB_MSGBLOCK [MSB$L_TEXT] = NML$_QIOBFOVF;
;  260     0258  3      NML$BLD_REPLY (NML$AB_MSGBLOCK, MSGSIZE);     ! Build message
;  261     0259  3      $SIGNAL_MSG (NML$AB_SNDBUFFER, .MSGSIZE);     ! Signal it.
;  262     0260  2      END;
;  263     0261  2
;  264     0262  2  PTR = .P2DSC [DSC$A_POINTER] + 4;       ! Skip over return count
;  265     0263  2
;  266     0264  2  !
;  267     0265  2  ! Add first search key value to the P2 buffer.
;  268     0266  2  !
;  269     0267  2  SELECTONE .LEN1 OF
;  270     0268  2      SET
;  271     0269  2      [-2]: PTR = CH$MOVE (2, ADR1, .PTR);            ! It's a word
;  272     0270  2      [0]: PTR = CH$MOVE (4, ADR1, .PTR);             ! It's a longword
;  273     0271  2      [1 TO 255]:                                     ! It's a string.
;  274     0272  3          BEGIN
;  275     0273  3          CH$WCHAR_A (.LEN1<0,8>, PTR);
;  276     0274  3          CH$WCHAR_A (.LEN1<8,8>, PTR);
;  277     0275  3          PTR = CH$MOVE (.LEN1, .ADR1, .PTR);
;  278     0276  3          END
;  279     0277  2      TES;
;  280     0278  2
;  281     0279  2  !
;  282     0280  2  ! Add search key two to buffer.
;  283     0281  2  !
;  284     0282  2  SELECTONE .LEN2 OF
;  285     0283  2      SET
;  286     0284  2      [-2]: PTR = CH$MOVE (2, ADR2, .PTR);            ! It's a word
;  287     0285  2      [0]: PTR = CH$MOVE (4, ADR2, .PTR);             ! It's a longword
;  288     0286  2      [1 TO 255]:                                     ! It's a string.
;  289     0287  3          BEGIN
;  290     0288  3          CH$WCHAR_A (.LEN2<0,8>, PTR);
;  291     0289  3          CH$WCHAR_A (.LEN2<8,8>, PTR);
;  292     0290  3          PTR = CH$MOVE (.LEN2, .ADR2, .PTR);
;  293     0291  3          END
;  294     0292  2      TES;
```

```
;  295          0293  2
;  296          0294  2 !
;  297          0295  2 ! Set up a context area of a string of nulls.  NETACP will
;  298          0296  2 ! replace the null string with a start value of the last database
;  299          0297  2 ! entry matched by the search key.  This allows NML to reissue
;  300          0298  2 ! the QIO so that NETACP will start searching where it left
;  301          0299  2 ! off from the previous QIO.  Used for plural entity operations
;  302          0300  2 ! (KNOWN, ACTIVE).
;  303          0301  2 !
;  304          0302  2 PTR = CH$MOVE ( NFB$C_CTX_SIZE, COLLATE_START_VALUE, .PTR);
;  305          0303  2 !
;  306          0304  2 ! Set up resulting descriptor for return.
;  307          0305  2 !
;  308          0306  2 RESDSC [DSC$W_LENGTH] = .PTR - .P2DSC [DSC$A_POINTER];
;  309          0307  2 RESDSC [DSC$A_POINTER] = .P2DSC [DSC$A_POINTER];
;  310          0308  2
;  311          0309  1 END;                                    ! End of NML$BLDP2

                                                        .TITLE  NML$UTILITY NML Utility routines
                                                        .IDENT  \V04-000\

                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                                        00000068  00000 P.AAA:  .LONG   104                                     ;
                                        00000000' 00004         .ADDRESS P2BUFFER                               ;
                                        00000100  00008 P.AAB:  .LONG   256                                     ;
                                        00000000' 0000C         .ADDRESS PRMBUFFER                              ;

                                                        .PSECT  $OWN$,NOEXE,2

                                                00000 P2BUFFER:
                                                        .BLKB   104
                                                00068 PRMBUFFER:
                                                        .BLKB   256
                                                00168 MSGLENGTH:
                                                        .BLKB   4
                                             00# 0016C COLLATE_START_VALUE:
                                                        .BYTE   0[64]                                           ;

                                                        .PSECT  $GLOBAL$,NOEXE,2

                                                00000 NML$GW_VOL_EXEC_ADDR::
                                                        .BLKB   2
                                                00002 NML$GW_PERM_EXEC_ADDR::
                                                        .BLKB   2
                                                00004 NML$T_VOL_EXEC_NAME::
                                                        .BLKB   16
                                        00000000  00014 NML$GQ_VOL_EXEC_NAME_DSC::
                                                        .LONG   0                                               ;
                                        00000000' 00018         .ADDRESS NML$T_VOL_EXEC_NAME                    ;
                                                0001C NML$T_PERM_EXEC_NAME::
                                                        .BLKB   16
                                        00000000  0002C NML$GQ_PERM_EXEC_NAME_DSC::
                                                        .LONG   0                                               ;
                                        00000000' 00030         .ADDRESS NML$T_PERM_EXEC_NAME                   ;
```

```
                                P2BFDSC=            P.AAA
                                PRMDSC=             P.AAB
                                        .EXTRN  NML$GB_EVTSRCTYP
                                        .EXTRN  NML$GQ_EVTSRCDSC
                                        .EXTRN  NML$GW_EVTCLASS
                                        .EXTRN  NML$GB_EVTMSKTYP
                                        .EXTRN  NML$GQ_EVTMSKDSC
                                        .EXTRN  NML$GW_EVTSNKADR
                                        .EXTRN  NML$GW_ACP_CHAN
                                        .EXTRN  NML$GL_LOGMASK, NML$GQ_ENTSTRDSC
                                        .EXTRN  NML$AB_QIOBUFFER
                                        .EXTRN  NML$GQ_QIOBFDSC
                                        .EXTRN  NML$AB_EXEBUFFER
                                        .EXTRN  NML$GL_EXEDATPTR
                                        .EXTRN  NML$GQ_EXEDATDSC
                                        .EXTRN  NML$GQ_EXEBFDSC
                                        .EXTRN  NML$AB_RCVBUFFER
                                        .EXTRN  NML$GQ_RCVBFDSC
                                        .EXTRN  NML$AB_SNDBUFFER
                                        .EXTRN  NML$GQ_SNDBFDSC
                                        .EXTRN  NML$GL_RCVDATLEN
                                        .EXTRN  NML$AB_CPTABLE, NML$AB_MSGBLOCK
                                        .EXTRN  NML$AB_ENTITY_ID
                                        .EXTRN  NML$AB_QUALIFIER_ID
                                        .EXTRN  NML$AB_ENTITYDATA
                                        .EXTRN  NML$AB_NML_NMV, NML$AB_PRMSEM
                                        .EXTRN  NML$AB_RECBUF, NML$AL_ENTINFTAB
                                        .EXTRN  NML$AL_PERMINFTAB
                                        .EXTRN  NML$AW_PRM_DES, NML$GB_CMD_VER
                                        .EXTRN  NML$GB_ENTITY_CODE
                                        .EXTRN  NML$GB_ENTITY_FORMAT
                                        .EXTRN  NML$GL_QUALIFIER_PST
                                        .EXTRN  NML$GB_QUALIFIER_FORMAT
                                        .EXTRN  NML$GB_FUNCTION
                                        .EXTRN  NML$GB_INFO, NML$GB_OPTIONS
                                        .EXTRN  NML$GL_PRMCODE, NML$GL_PRS_FLGS
                                        .EXTRN  NML$GL_NML_ENTITY
                                        .EXTRN  NML$GQ_NETNAMDSC
                                        .EXTRN  NML$GQ_RECBFDSC
                                        .EXTRN  NML$GW_PRMDESCNT
                                        .EXTRN  NML$GB_NCP_VERSION
                                        .EXTRN  NML$_QIOBFOVF, NML$BLD_REPLY
                                        .EXTRN  NML$OPENFILE, NMA$SEARCHFLD
                                        .EXTRN  NML$ERROR_1, NML$READRECORD
                                        .EXTRN  NML$NETQIO

                                        .PSECT  $CODE$,NOWRT,2

                            01FC 00000  .ENTRY  NML$BLDP2, Save R2,R3,R4,R5,R6,R7,R8        ; 0180
                58 00000000G  00  9E 00002  MOVAB   NML$AB_MSGBLOCK, R8                     ;
                5E            04  C2 00009  SUBL2   #4, SP                                ;
                50            04  D0 0000C  MOVL    #4, COUNT                             ; 0234
                52      04    AC  D0 0000F  MOVL    LEN1, R2                              ; 0235
FFFFFFFE        8F            52  D1 00013  CMPL    R2, #-2                               ; 0237
                              05  12 0001A  BNEQ    1$                                    ;
                50            02  C0 0001C  ADDL2   #2, COUNT                            ;
                              19  11 0001F  BRB     3$                                    ;
```

```
                                                 52  D5 00021 1$:     TSTL    R2                                          ; 0238
                                                 05  12 00023        BNEQ    2$
                                      50         04  C0 00025        ADDL2   #4, COUNT
                                                 10  11 00028        BRB     3$
                                                 0E  15 0002A 2$:     BLEQ    3$                                          ; 0239
                                 000000FF    8F  52  D1 0002C        CMPL    R2, #255
                                                 05  14 00033        BGTR    3$
                                      50     02 A240  9E 00035        MOVAB   2(R2)[COUNT], COUNT
                                      57     0C   AC  D0 0003A 3$:    MOVL    LEN2, R7                                    ; 0242
                                 FFFFFFFE    8F  57  D1 0003E        CMPL    R7, #-2                                     ; 0244
                                                 05  12 00045        BNEQ    4$
                                      50         02  C0 00047        ADDL2   #2, COUNT
                                                 19  11 0004A        BRB     6$
                                                 57  D5 0004C 4$:     TSTL    R7                                          ; 0245
                                                 05  12 0004E        BNEQ    5$
                                      50         04  C0 00050        ADDL2   #4, COUNT
                                                 10  11 00053        BRB     6$
                                                 0E  15 00055 5$:     BLEQ    6$                                          ; 0246
                                 000000FF    8F  57  D1 00057        CMPL    R7, #255
                                                 05  14 0005E        BGTR    6$
                                      50     02 A740  9E 00060        MOVAB   2(R7)[COUNT], COUNT
                                      50     40   A0  9E 00065 6$:    MOVAB   64(R0), COUNT                               ; 0249
                                      56     14   AC  D0 00069        MOVL    P2DSC, R6                                   ; 0250
                50          66             10       00  ED 0006D        CMPZV   #0, #16, (R6), COUNT
                                                 2F  18 00072        BGEQ    7$
                                      68         04  D0 00074        MOVL    #4, NML$AB_MSGBLOCK                         ; 0255
                                   04    A8      05  8E 00077        MNEGB   #5, NML$AB_MSGBLOCK+4                       ; 0256
                                   0C    A8 00000000G  8F  D0 0007B   MOVL    #NML$_QIOBFOVF, NML$AB_MSGBLOCK+12          ; 0257
                                            4100     8F  BB 00083        PUSHR   #^M<R8,SP>                                  ; 0258
                                 00000000G   00         02  FB 00087        CALLS   #2, NML$BLD_REPLY
                                                 6E  DD 0008E        PUSHL   MSGSIZE                                     ; 0259
                                      00000000G  00  9F 00090        PUSHAB  NML$AB_SNDBUFFER
                                      01F90000   8F  DD 00096        PUSHL   #33095680
                                 00000000G   00         03  FB 0009C        CALLS   #3, LIB$SIGNAL
                50          04          A6       04  C1 000A3 7$:     ADDL3   #4, 4(R6), PTR                              ; 0262
                            FFFFFFFE    8F  52  D1 000A8        CMPL    R2, #-2                                     ; 0269
                                                 06  12 000AF        BNEQ    8$
                                      83         08   AC  B0 000B1        MOVW    ADR1, (PTR)+
                                                 1E  11 000B5        BRB     10$
                                                 52  D5 000B7 8$:     TSTL    R2                                          ; 0270
                                                 06  12 000B9        BNEQ    9$
                                      83         08   AC  D0 000BB        MOVL    ADR1, (PTR)+
                                                 14  11 000BF        BRB     10$
                                                 12  15 000C1 9$:     BLEQ    10$                                         ; 0271
                                 000000FF    8F  52  D1 000C3        CMPL    R2, #255
                                                 09  14 000CA        BGTR    10$
                                      83         04   AC  B0 000CC        MOVW    LEN1, (PTR)+                            ; 0273
                63          08          BC       52  28 000D0        MOVC3   R2, @ADR1, (PTR)                            ; 0275
                            FFFFFFFE    8F  57  D1 000D5 10$:    CMPL    R7, #-2                                     ; 0284
                                                 06  12 000DC        BNEQ    11$
                                      83         10   AC  B0 000DE        MOVW    ADR2, (PTR)+
                                                 1E  11 000E2        BRB     13$
                                                 57  D5 000E4 11$:     TSTL    R7                                          ; 0285
                                                 06  12 000E6        BNEQ    12$
                                      83         10   AC  D0 000E8        MOVL    ADR2, (PTR)+
                                                 14  11 000EC        BRB     13$
                                                 12  15 000EE 12$:     BLEQ    13$                                         ; 0286
```

```
                    000000FF    8F              57  D1 000F0          CMPL    R7, #255                                    ;
                                                09  14 000F7          BGTR    13$                                         ;
                                83      0C      AC  B0 000F9          MOVW    LEN2, (PTR)+                                ; 0288
                63          10  BC              57  28 000FD          MOVC3   R7, @ADR2, (PTR)                            ; 0290
                63 00C00000'    00      0040    8F  28 00102 13$:     MOVC3   #64, COLLATE_START_VALUE, (PTR)             ; 0302
                                50      18      AC  D0 0010C          MOVL    RESDSC, R0                                  ; 0306
                60              53      04      A6  A3 00110          SUBW3   4(R6), PTR, (R0)                            ;
                            04  A0      04      A6  D0 00115          MOVL    4(R6), 4(R0)                                ; 0307
                                                    04 0011A          RET                                                 ; 0309

; Routine Size:  283 bytes,    Routine Base:  $CODE$ + 0000
```

```
;   313          0310  1 %SBTTL 'NML$CHKEXE  Check node address against executor'
;   314          0311  1 GLOBAL ROUTINE NML$CHKEXE (NODE_ID, NODE_ADDR, NODE_NAME_LEN, NODE_NAME_ADDR) =
;   315          0312  1
;   316          0313  1 !++
;   317          0314  1 ! FUNCTIONAL DESCRIPTION:
;   318          0315  1 !
;   319          0316  1 !       This routine compares the specified node address with the executor node
;   320          0317  1 !       address to see if they match.
;   321          0318  1 !
;   322          0319  1 ! FORMAL PARAMETERS:
;   323          0320  1 !       NODE_ID         Equals NMA$C_PCNO_ADD if routine is to check the executor
;   324          0321  1 !                       address and NMA$C_PCNO_NNA if the routine is to check
;   325          0322  1 !                       the executor's name.
;   326          0323  1 !       NODE_ADDR       Node address (word) to match against executor's
;   327          0324  1 !       NODE_NAME_LEN   Length of node name to match against executor's
;   328          0325  1 !       NODE_NAME_ADDR  Address of node name string to match against executor's
;   329          0326  1 !
;   330          0327  1 ! ROUTINE VALUE:
;   331          0328  1 ! COMPLETION CODES:
;   332          0329  1 !       nml$_sts_cmp - The node id is not the executor's
;   333          0330  1 !       nml$_sts_suc - The node id is the executor's
;   334          0331  1 !
;   335          0332  1 !--
;   336          0333  1
;   337          0334  2 BEGIN
;   338          0335  2
;   339          0336  2 MAP
;   340          0337  2     node_id     : WORD,
;   341          0338  2     node_addr   : WORD;
;   342          0339  2
;   343          0340  2 LOCAL
;   344          0341  2     exeadr      : WORD,
;   345          0342  2     exenambuf   : VECTOR [6, BYTE],
;   346          0343  2     exenamdsc   : DESCRIPTOR,
;   347          0344  2     exenamlen,
;   348          0345  2     status;
;   349          0346  2
;   350          0347  2 MAP
;   351          0348  2     nml$gb_options : BBLOCK [1];
;   352          0349  2
;   353          0350  2 !
;   354          0351  2 ! If this is a permanent database operation, and the node permanent
;   355          0352  2 ! data base file isn't already open, open it.
;   356          0353  2 !
;   357          0354  2 IF .nml$gb_options [nma$v_opt_per] THEN
;   358          0355  2     nml$openfile (nma$c_opn_node, nma$c_opn_ac_ro);
;   359          0356  2 status = nml$_sts_cmp;
;   360          0357  2
;   361          0358  2 !
;   362          0359  2 ! If this routine was called to compare a node name against the executor's
;   363          0360  2 ! name, call NML$GETEXENAM to do the comparison.
;   364          0361  2 !
;   365          0362  2 IF .node_id EQL nma$c_pcno_nna THEN
;   366          0363  3     BEGIN
;   367          0364  3     exenamdsc [dsc$w_length] = 6;
;   368          0365  3     exenamdsc [dsc$a_pointer] = exenambuf;
;   369          0366  3     IF nml$getexenam (exenamdsc, exenamlen) THEN
```

```
;  370          0367  3          IF CH$EQL (.node_name_len, .node_name_addr, .exenamlen, exenambuf) THEN
;  371          0368  3              status = nml$_sts_suc;
;  372          0369  3      END
;  373          0370  2  ELSE
;  374          0371  2      !
;  375          0372  2      ! If this routine was called to compare a node address against the executor's
;  376          0373  2      ! address, call NML$GETEXEADR to do the comparison.
;  377          0374  2      !
;  378          0375  3      BEGIN
;  379          0376  3      IF .node_addr EQL 0 THEN
;  380          0377  3          status = nml$_sts_suc
;  381          0378  3      ELSE
;  382          0379  4          BEGIN
;  383          0380  4          IF nml$getexeadr (exeadr) THEN
;  384          0381  4              IF .exeadr EQL .node_addr THEN
;  385          0382  4                  status = nml$_sts_suc;
;  386          0383  3          END;
;  387          0384  2      END;
;  388          0385  2  RETURN .status
;  389          0386  2
;  390          0387  1  END;                               ! End of nml$chkexe


                                    001C 00000          .ENTRY  NML$CHKEXE, Save R2,R3,R4                       ; 0311
                              5E    18   C2 00002          SUBL2   #24, SP                                         ;
                 00000000G    00   95 00005          TSTB    NML$GB_OPTIONS                                  ; 0354
                                    09   18 0000B          BGEQ    1$                                              ;
                                    7E   7C 0000D          CLRQ    -(SP)                                           ; 0355
                 00000000G    00   02   FB 0000F          CALLS   #2, NML$OPENFILE                                ;
                              54    10   CE 00016  1$:      MNEGL   #16, STATUS                                     ; 0356
                    01F4      8F         04   AC   B1 00019          CMPW    NODE_ID, #500                           ; 0362
                                    23   12 0001F          BNEQ    2$                                              ;
                       08     AE         06   B0 00021          MOVW    #6, EXENAMDSC                           ; 0364
                       0C     AE         10   AE   9E 00025          MOVAB   EXENAMBUF, EXENAMDSC+4                  ; 0365
                                    5E   DD 0002A          PUSHL   SP                                              ; 0366
                                         0C   AE   9F 0002C          PUSHAB  EXENAMDSC                               ;
                 00000000V    00   02   FB 0002F          CALLS   #2, NML$GETEXENAM                               ;
                              27    50   E9 00036          BLBC    R0, 5$                                          ;
          6E          00      10   BC   0C   AC   2D 00039          CMPC5   NODE_NAME_LEN, @NODE_NAME_ADDR, #0, -   ; 0367
                                         10   AE        00040                  EXENAMLEN, EXENAMBUF                    ;
                                    17   11 00042          BRB     3$                                              ;
                                         08   AC   B5 00044  2$:     TSTW    NODE_ADDR                               ; 0376
                                    14   13 00047          BEQL    4$                                              ;
                                         04   AE   9F 00049          PUSHAB  EXEADR                                  ; 0380
                 00000000V    00   01   FB 0004C          CALLS   #1, NML$GETEXEADR                               ;
                              0A    50   E9 00053          BLBC    R0, 5$                                          ;
                       08     AC         04   AE   B1 00056          CMPW    EXEADR, NODE_ADDR                       ; 0381
                                    03   12 0005B  3$:      BNEQ    5$                                              ;
                              54    01   D0 0005D  4$:      MOVL    #1, STATUS                                      ; 0382
                              50    54   D0 00060  5$:      MOVL    STATUS, R0                                      ; 0385
                                    04   00063          RET                                                     ; 0387

; Routine Size:  100 bytes,    Routine Base:  $CODE$ + 011B
```

```
;  392          0388  1 %SBTTL 'NML$SET_UP_EXEC_ID   Set up globals for executors name and address'
;  393          0389  1 GLOBAL ROUTINE NML$SET_UP_EXEC_ID =
;  394          0390  1
;  395          0391  1 !++
;  396          0392  1 ! FUNCTIONAL DESCRIPTION:
;  397          0393  1 !       This routine gets the name and address of the executor node from the
;  398          0394  1 !       volatile or permanent database and saves them in some global fields.
;  399          0395  1 !
;  400          0396  1 ! FORMAL PARAMETERS:
;  401          0397  1 !       NONE
;  402          0398  1 !
;  403          0399  1 ! ROUTINE VALUE:
;  404          0400  1 ! COMPLETION CODES:
;  405          0401  1 !
;  406          0402  1 !--
;  407          0403  1
;  408          0404  2 BEGIN
;  409          0405  2
;  410          0406  2 MAP
;  411          0407  2     nml$gb_options : BBLOCK [1];
;  412          0408  2
;  413          0409  2 LOCAL
;  414          0410  2     fldadr,
;  415          0411  2     fldsize,
;  416          0412  2     temp,
;  417          0413  2     recdsc :    VECTOR [2],
;  418          0414  2     p2dsc  :    VECTOR [2],     ! Descriptor for QIO P2 (Key) buffer.
;  419          0415  2     ptr,
;  420          0416  2     status;
;  421          0417  2
;  422        P 0418  2 $NFBDSC(NFBDSC, SHOW, , LNI
;  423        P 0419  2         ,NFB$C_WILDCARD,          ! Search key 1 = wildcard, oper1 = eql
;  424        P 0420  2         ,NFB$C_WILDCARD,          ! Search key 2 = wildcard, oper2 = eql
;  425        P 0421  2         ,ADD
;  426          0422  2         ,NAM);
;  427          0423  2
;  428          0424  2 IF .nml$gb_options [nma$v_opt_per] THEN
;  429          0425  3     BEGIN
;  430          0426  3     IF .nml$gw_perm_exec_addr EQL 0 THEN
;  431          0427  4         BEGIN
;  432          0428  4         !
;  433          0429  4         ! If the node permanent data base file isn't already open, open it.
;  434          0430  4         !
;  435          0431  4         nml$openfile (nma$c_opn_node, nma$c_opn_ac_ro);
;  436          0432  4         recdsc [1] = .prmdsc [1];
;  437          0433  4         status = nml$readrecord (nma$c_opn_node,          ! Node database file ID
;  438          0434  4                                  UPLIT (nmn$c_typ_key_ref),   ! ISAM key = node type
;  439          0435  4                                  UPLIT (nmn$c_typ_key_len,    ! ISAM key value = executor
;  440          0436  4                                  UPLIT (nml$c_executor)),
;  441          0437  4                                  prmdsc,                  ! Read buffer descriptor
;  442          0438  4                                  recdsc,                  ! Return data descriptor
;  443          0439  4                                  temp);                   ! Not used.
;  444          0440  4         IF .status THEN
;  445          0441  5             BEGIN
;  446          0442  5             fldadr = 0;
;  447          0443  5             IF nma$searchfld (  recdsc,
;  448          0444  5                                 nma$c_pcno_add,
```

```
;  449          0445  5                                   fldsize,
;  450          0446  5                                   fldadr) THEN
;  451          0447  5                       CH$MOVE (2, CH$PTR (.fldadr), nml$gw_perm_exec_addr);
;  452          0448  5                   fldadr = 0;
;  453          0449  5                   IF nma$searchfld (  recdsc,
;  454          0450  5                                       nma$c_pcno_nna,
;  455          0451  5                                       nml$gq_perm_exec_name_dsc [0],
;  456          0452  5                                       fldadr) THEN
;  457          0453  5                       CH$MOVE (.nml$gq_perm_exec_name_dsc [0],
;  458          0454  5                                .fldadr,
;  459          0455  5                                .nml$gq_perm_exec_name_dsc [1]);
;  460          0456  4                   END;
;  461          0457  3               END;
;  462          0458  3           END
;  463          0459  2       ELSE
;  464          0460  3           BEGIN
;  465          0461  3           IF .nml$gw_vol_exec_addr EQL 0 THEN
;  466          0462  4               BEGIN
;  467          0463  4               !
;  468          0464  4               ! Set up search and start key buffer to get any entry in the data
;  469          0465  4               ! base.  The executor node database only has one entry.  Then issue
;  470          0466  4               ! the QIO to get the node address.
;  471          0467  4               !
;  472          0468  4               nml$bldp2 ( -1, 0, -1, 0, p2bfdsc, p2dsc);
;  473          0469  4               status = nml$netqio (   nfbdsc,
;  474          0470  4                                       p2dsc,
;  475          0471  4                                       fldsize,
;  476          0472  4                                       prmdsc);
;  477          0473  4               IF .status THEN
;  478          0474  5                   BEGIN
;  479          0475  5                   ptr = .prmdsc [1];
;  480          0476  5                   CH$MOVE (2, .ptr, nml$gw_vol_exec_addr);
;  481          0477  5                   ptr = .ptr + 4;
;  482          0478  5                   CH$COPY (2, .ptr, 0, 4, nml$gq_vol_exec_name_dsc [0]);
;  483          0479  5                   CH$MOVE (.nml$gq_vol_exec_name_dsc [0],
;  484          0480  5                            .ptr + 2,
;  485          0481  5                            .nml$gq_vol_exec_name_dsc [1]);
;  486          0482  5                   RETURN nml$_sts_suc
;  487          0483  4                   END;
;  488          0484  3               END;
;  489          0485  2           END;
;  490          0486  2       RETURN .status;
;  491          0487  1       END;                            ! End of nml$set_up_exec_id


                                                              .PSECT  $PLIT$,NOWRT,NOEXE,2

                                00000020  00010 P.AAC:        .LONG   32                              ;
                                00000000' 00014               .ADDRESS U.1                            ;
                                00000001  00018 P.AAD:        .LONG   1                               ;
                                00000007  0001C P.AAF:        .LONG   7                               ;
                                00000002  00020 P.AAE:        .LONG   2                               ;
                                00000000' 00024               .ADDRESS P.AAF                          ;

                                                              .PSECT  $OWN$,NOEXE,2
```

```
                                22  001AC ;_NFB
                                       U.1:     .BYTE   34                                                              ;
                                    00  001AD            .BYTE   0                                                       ;
                                    01  001AE            .BYTE   1                                                       ;
                                    00  001AF            .BYTE   0                                                       ;
                              00000001  001B0            .LONG   1                                                       ;
                              00000001  001B4            .LONG   1                                                       ;
                                    00  001B8            .BYTE   0                                                       ;
                                    00  001B9            .BYTE   0                                                       ;
                                  0000  001BA            .WORD   0                                                       ;
                              01010010  001BC            .LONG   16842768                                                ;
                              01020041  001C0            .LONG   16908353                                                ;
                              00000000  001C4            .LONG   0                                                       ;
                                        001C8            .BLKB   4

                                                 U.2=                P.AAC

                                                         .PSECT  $CODE$,NOWRT,2

                                    07FC 00000           .ENTRY  NML$SET_UP_EXEC_ID, Save R2,R3,R4,R5,R6,R7,-;  0389
                                                                 R8,R9,R10                                       ;
                          5A 00000000G   00   9E 00002           MOVAB   NMA$SEARCHFLD, R10                      ;
                          59 00000000'   00   9E 00009           MOVAB   PRMDSC+4, R9                            ;
                          58 00000000'   00   9E 00010           MOVAB   NML$GW_PERM_EXEC_ADDR, R8               ;
                          5E             1C   C2 00017           SUBL2   #28, SP                                 ;
                             00000000G   00   95 0001A           TSTB    NML$GB_OPTIONS                          ;  0424
                                         6C   18 00020           BGEQ    2$                                      ;
                                         68   B5 00022           TSTW    NML$GW_PERM_EXEC_ADDR                   ;  0426
                                         6B   12 00024           BNEQ    3$                                      ;
                                         7E   7C 00026           CLRQ    -(SP)                                   ;  0431
                          00000000G      00   02 FB 00028           CALLS   #2, NML$OPENFILE                        ;
                                18       AE   69 D0 0002F           MOVL    PRMDSC+4, RECDSC+4                      ;  0432
                                         5E   DD 00033           PUSHL   SP                                      ;  0433
                                      18 AE   9F 00035           PUSHAB  RECDSC                                  ;
                                      FC A9   9F 00038           PUSHAB  PRMDSC                                  ;
                                      14 A9   9F 0003B           PUSHAB  P.AAE                                   ;  0435
                                      0C A9   9F 0003E           PUSHAB  P.AAD                                   ;  0434
                                         7E   D4 00041           CLRL    -(SP)                                   ;  0433
                          00000000G      00   06 FB 00043           CALLS   #6, NML$READRECORD                      ;
                                         57   50 D0 0004A           MOVL    R0, STATUS                              ;
                                         6E   57 E9 0004D           BLBC    STATUS, 4$                              ;  0440
                                      04 AE   D4 00050           CLRL    FLDADR                                  ;  0442
                                      04 AE   9F 00053           PUSHAB  FLDADR                                  ;  0443
                                      0C AE   9F 00056           PUSHAB  FLDSIZE                                 ;
                                   7E 01F6    8F 3C 00059           MOVZWL  #502, -(SP)                             ;
                                      20 AE   9F 0005E           PUSHAB  RECDSC                                  ;
                                   6A         04 FB 00061           CALLS   #4, NMA$SEARCHFLD                       ;
                                   04         50 E9 00064           BLBC    R0, 1$                                  ;
                                   68      04 BE B0 00067           MOVW    @FLDADR, NML$GW_PERM_EXEC_ADDR          ;  0447
                                      04 AE   D4 0006B 1$:         CLRL    FLDADR                                  ;  0448
                                      04 AE   9F 0006E           PUSHAB  FLDADR                                  ;  0449
                                      2A A8   9F 00071           PUSHAB  NML$GQ_PERM_EXEC_NAME_DSC               ;  0451
                                   7E 01F4    8F 3C 00074           MOVZWL  #500, -(SP)                             ;  0449
                                      20 AE   9F 00079           PUSHAB  RECDSC                                  ;
                                   6A         04 FB 0007C           CALLS   #4, NMA$SEARCHFLD                       ;
                                   5E         50 E9 0007F           BLBC    R0, 5$                                  ;
```

```
                                50      2E   A8  D0 00082          MOVL    NML$GQ_PERM_EXEC_NAME_DSC+4, R0                          ; 0455
                          60    04      BE   2A  A8  28 00086          MOVC3   NML$GQ_PERM_EXEC_NAME_DSC, @FLDADR, (R0)
                                                  52  11 0008C          BRB     5$                                               ; 0424
                                        FE   A8  B5 0008E  2$:     TSTW    NML$GW_VOL_EXEC_ADDR                             ; 0461
                                                  4D  12 00091  3$:     BNEQ    5$
                                        0C   AE  9F 00093          PUSHAB  P2DSC                                            ; 0468
                                        F4   A9  9F 00096          PUSHAB  P2BFDSC
                                                  7E  D4 00099          CLRL    -(SP)
                                50                01  CE 0009B          MNEGL   #1, -(SP)
                                                  7E  D4 0009E          CLRL    -(SP)
                                7E                01  CE 000A0          MNEGL   #1, -(SP)
                             FDD9   CF            06  FB 000A3          CALLS   #6, NML$BLDP2
                                        FC   A9  9F 000A8          PUSHAB  PRMDSC                                           ; 0469
                                        0C   AE  9F 000AB          PUSHAB  FLDSIZE
                                        14   AE  9F 000AE          PUSHAB  P2DSC
                                        04   A9  9F 000B1          PUSHAB  NFBDSC
                        00000000G       00        04  FB 000B4          CALLS   #4, NML$NETQIO
                                57                50  D0 000BB          MOVL    R0, STATUS
                                1F                57  E9 000BE  4$:     BLBC    STATUS, 5$                               ; 0473
                                56                69  D0 000C1          MOVL    PRMDSC+4, PTR                            ; 0475
                                   FE   A8        86  B0 000C4          MOVW    (PTR)+, NML$GW_VOL_EXEC_ADDR             ; 0476
                                56                02  C0 000C8          ADDL2   #2, PTR                                  ; 0477
                04              00                02  2C 000CB          MOVC5   #2, (PTR), #0, #4, NML$GQ_VOL_EXEC_NAME_DSC ; 0478
                                                  12  A8       000D0
                                50                16  A8  D0 000D2          MOVL    NML$GQ_VOL_EXEC_NAME_DSC+4, R0           ; 0481
                          60    02      A6        12  A8  28 000D6          MOVC3   NML$GQ_VOL_EXEC_NAME_DSC, 2(PTR), (R0)
                                50                01  D0 000DC          MOVL    #1, R0                                   ; 0482
                                                  04  000DF          RET
                                50                57  D0 000E0  5$:     MOVL    STATUS, R0                               ; 0486
                                                  04  000E3          RET                                              ; 0487

; Routine Size:  228 bytes,    Routine Base:  $CODE$ + 017F
```

```
;  493          0488  1 %SBTTL 'NML$GETEXEADR  Get executor node address'
;  494          0489  1 GLOBAL ROUTINE NML$GETEXEADR (ADDR) =
;  495          0490  1
;  496          0491  1 !++
;  497          0492  1 ! FUNCTIONAL DESCRIPTION:
;  498          0493  1 !
;  499          0494  1 !       This routine returns the executor node address.
;  500          0495  1 !
;  501          0496  1 ! FORMAL PARAMETERS:
;  502          0497  1 !       ADDR            Address of word to contain node address.
;  503          0498  1 !
;  504          0499  1 ! IMPLICIT INPUTS:
;  505          0500  1 !
;  506          0501  1 !       NML$GB_OPTIONS contains the command message options.
;  507          0502  1 !
;  508          0503  1 !       If this is a permanent data base operation then it is assumed
;  509          0504  1 !       that the node file is already open.
;  510          0505  1 !
;  511          0506  1 ! IMPLICIT OUTPUTS:
;  512          0507  1 !
;  513          0508  1 !       NONE
;  514          0509  1 !
;  515          0510  1 ! ROUTINE VALUE:
;  516          0511  1 ! COMPLETION CODES:
;  517          0512  1 !
;  518          0513  1 !       If the executor node address is found then success (NML$_STS_SUC) is
;  519          0514  1 !       returned.  If the node address is not found, then a zero address is
;  520          0515  1 !       returned along with failure (NML$_STS_PTY).
;  521          0516  1 !
;  522          0517  1 ! SIDE EFFECTS:
;  523          0518  1 !
;  524          0519  1 !       Destroys contents of PRMBUFFER.
;  525          0520  1 !
;  526          0521  1 !--
;  527          0522  1
;  528          0523  2 BEGIN
;  529          0524  2
;  530          0525  2 MAP
;  531          0526  2     nml$gb_options : BBLOCK [1];
;  532          0527  2
;  533          0528  2 LOCAL
;  534          0529  2     exec_addr,
;  535          0530  2     status;
;  536          0531  2
;  537          0532  2 IF .nml$gb_options [nma$v_opt_per] THEN
;  538          0533  2     exec_addr = nml$gw_perm_exec_addr
;  539          0534  2 ELSE
;  540          0535  2     exec_addr = nml$gw_vol_exec_addr;
;  541          0536  2
;  542          0537  2 IF .(.exec_addr)<0,16> EQL 0 THEN
;  543          0538  3     BEGIN
;  544          0539  3     status = nml$set_up_exec_id ();
;  545          0540  3     IF NOT .status THEN
;  546          0541  3         !
;  547          0542  3         ! No executor entry found.  This should happen only for the permanent
;  548          0543  3         ! database, and there, not very often.
;  549          0544  3         !
```

```
;  550          0545  4          BEGIN
;  551          0546  4          (.addr)<0,16> = 0;
;  552          0547  4          RETURN nml$_sts_pty;
;  553          0548  3          END;
;  554          0549  2      END;
;  555          0550  2 CH$MOVE (2, .exec_addr, .addr);
;  556          0551  2 RETURN nml$_sts_suc;
;  557          0552  1 END;                              ! End of NML$GETEXEADR



                                        0004 00000          .ENTRY  NML$GETEXEADR, Save R2                  ; 0489
                            00000000G   00   95 00002       TSTB    NML$GB_OPTIONS                          ; 0532
                                        09   18 00008       BGEQ    1$                                      ;
                          52 00000000'  00   9E 0000A       MOVAB   NML$GW_PERM_EXEC_ADDR, EXEC_ADDR        ; 0533
                                        07   11 00011       BRB     2$                                      ;
                          52 00000000'  00   9E 00013 1$:   MOVAB   NML$GW_VOL_EXEC_ADDR, EXEC_ADDR        ; 0535
                                        62   B5 0001A 2$:   TSTW    (EXEC_ADDR)                             ; 0537
                                        0F   12 0001C       BNEQ    3$                                      ;
                      FEF9   CF         00   FB 0001E       CALLS   #0, NML$SET_UP_EXEC_ID                  ; 0539
                             07         50   E8 00023       BLBS    STATUS, 3$                              ; 0540
                                  04    BC   B4 00026       CLRW    @ADDR                                   ; 0546
                             50         0C   CE 00029       MNEGL   #12, R0                                 ; 0547
                                        04   0002C         RET                                             ;
                        04   BC         62   B0 0002D 3$:   MOVW    (EXEC_ADDR), @ADDR                      ; 0550
                             50         01   D0 00031       MOVL    #1, R0                                  ; 0551
                                        04   00034         RET                                             ; 0552

; Routine Size:  53 bytes,    Routine Base:  $CODE$ + 0263
```

```
;  559     0553  1 %SBTTL 'NML$GETEXENAM  Get executor node name'
;  560     0554  1 GLOBAL ROUTINE NML$GETEXENAM (BUFDSC, RESLEN) =
;  561     0555  1
;  562     0556  1 !++
;  563     0557  1 ! FUNCTIONAL DESCRIPTION:
;  564     0558  1 !
;  565     0559  1 !       This routine returns the executor node name.
;  566     0560  1 !
;  567     0561  1 ! FORMAL PARAMETERS:
;  568     0562  1 !
;  569     0563  1 !       BUFDSC          Address of descriptor of buffer to contain ASCII
;  570     0564  1 !                       node name.
;  571     0565  1 !       RESLEN          Resulting length of node name string.
;  572     0566  1 !
;  573     0567  1 ! IMPLICIT INPUTS:
;  574     0568  1 !
;  575     0569  1 !       If this is a permanent data base operation then it is assumed
;  576     0570  1 !       that the node file is already open.
;  577     0571  1 !
;  578     0572  1 ! IMPLICIT OUTPUTS:
;  579     0573  1 !
;  580     0574  1 !       NONE
;  581     0575  1 !
;  582     0576  1 ! ROUTINE VALUE:
;  583     0577  1 ! COMPLETION CODES:
;  584     0578  1 !
;  585     0579  1 !       If the executor node name is found then success (NML$_STS_SUC) is
;  586     0580  1 !       returned.  If the node name is not found a zero length counted string
;  587     0581  1 !       is returned along with failure (NML$_STS_PTY).
;  588     0582  1 !
;  589     0583  1 ! SIDE EFFECTS:
;  590     0584  1 !
;  591     0585  1 !       NONE
;  592     0586  1 !
;  593     0587  1 !--
;  594     0588  1
;  595     0589  2 BEGIN
;  596     0590  2
;  597     0591  2 MAP
;  598     0592  2     bufdsc : REF DESCRIPTOR,
;  599     0593  2     nml$gb_options : BBLOCK [1];
;  600     0594  2
;  601     0595  2 LOCAL
;  602     0596  2     exec_dsc_addr: REF VECTOR,
;  603     0597  2     status;
;  604     0598  2
;  605     0599  2 IF .nml$gb_options [nma$v_opt_per] THEN
;  606     0600  2     exec_dsc_addr = nml$gq_perm_exec_name_dsc
;  607     0601  2 ELSE
;  608     0602  2     exec_dsc_addr = nml$gq_vol_exec_name_dsc;
;  609     0603  2
;  610     0604  2 IF .exec_dsc_addr [0] EQL 0 THEN
;  611     0605  3     BEGIN
;  612     0606  3     status = nml$set_up_exec_id ();
;  613     0607  3     IF NOT .status THEN
;  614     0608  3         !
;  615     0609  3         ! No executor entry found.  This should happen only for the permanent
```

```
;  616          0610  3          ! database, and there, not very often.
;  617          0611  3          !
;  618          0612  4          BEGIN
;  619          0613  4          .reslen = 0;
;  620          0614  4          RETURN nml$_sts_pty;
;  621          0615  3          END;
;  622          0616  2      END;
;  623          0617  2  .reslen = .exec_dsc_addr [0];
;  624          0618  2  IF ..reslen LEQU .bufdsc [dsc$w_length] THEN
;  625          0619  3      BEGIN
;  626          0620  3      CH$MOVE (..reslen, .exec_dsc_addr [1], .bufdsc [dsc$a_pointer]);
;  627          0621  3      RETURN nml$_sts_suc;
;  628          0622  3      END
;  629          0623  2  ELSE
;  630          0624  2      RETURN nml$_sts_pty;
;  631          0625  1  END;                                  ! End of NML$GETEXENAM


                                                003C 00000         .ENTRY  NML$GETEXENAM, Save R2,R3,R4,R5                 ; 0554
                                    00000000G   00   95 00002      TSTB    NML$GB_OPTIONS                                  ; 0599
                                                09   18 00008      BGEQ    1$                                              ;
                                52 00000000'    00   9E 0000A      MOVAB   NML$GQ_PERM_EXEC_NAME_DSC, EXEC_DSC_ADDR        ; 0600
                                                07   11 00011      BRB     2$                                              ;
                                52 00000000'    00   9E 00013 1$:  MOVAB   NML$GQ_VOL_EXEC_NAME_DSC, EXEC_DSC_ADDR        ; 0602
                                                62   D5 0001A 2$:  TSTL    (EXEC_DSC_ADDR)                                 ; 0604
                                                0D   12 0001C      BNEQ    3$                                              ;
                                      FEC4      CF   00 0001E      CALLS   #0, NML$SET_UP_EXEC_ID                          ; 0606
                                                05   50 00023      BLBS    STATUS, 3$                                      ; 0607
                                           08   BC   D4 00026      CLRL    @RESLEN                                         ; 0613
                                                1B   11 00029      BRB     4$                                              ; 0614
                                           08   BC   62 D0 0002B 3$:  MOVL    (EXEC_DSC_ADDR), @RESLEN                    ; 0617
                                                50   04 AC D0 0002F      MOVL    BUFDSC, R0                                  ; 0618
                  08   BC             60           10   00 ED 00033      CMPZV   #0, #16, (R0), @RESLEN                      ;
                                                     0B   1F 00039      BLSSU   4$                                          ;
                                04   B0       04   B2   08   BC 28 0003B      MOVC3   @RESLEN, @4(EXEC_DSC_ADDR), @4(R0)  ; 0620
                                                50   01 D0 00042      MOVL    #1, R0                                      ; 0624
                                                     04 00045      RET                                                 ;
                                                50   0C   CE 00046 4$:  MNEGL   #12, R0                                    ;
                                                     04 00049      RET                                                 ; 0625

; Routine Size:  74 bytes,    Routine Base:  $CODE$ + 0298
```

```
;   633          0626  1 %SBTTL 'NML$GETNODNAM  Get node name given the address'
;   634          0627  1 GLOBAL ROUTINE NML$GETNODNAM (ADDR, BUFDSC, RESLEN) =
;   635          0628  1
;   636          0629  1 !++
;   637          0630  1 ! FUNCTIONAL DESCRIPTION:
;   638          0631  1 !
;   639          0632  1 !       This routine returns the node name that matches the
;   640          0633  1 !       specified address.
;   641          0634  1 !
;   642          0635  1 ! FORMAL PARAMETERS:
;   643          0636  1 !
;   644          0637  1 !       ADDR            Node address.
;   645          0638  1 !       BUFDSC          Address of descriptor of buffer to contain ASCII
;   646          0639  1 !                       node name.
;   647          0640  1 !       RESLEN          Resulting length of node name string.
;   648          0641  1 !
;   649          0642  1 ! IMPLICIT INPUTS:
;   650          0643  1 !
;   651          0644  1 !       NML$GB_OPTIONS contains the command message options.
;   652          0645  1 !
;   653          0646  1 !       If this is a permanent data base operation then it is assumed
;   654          0647  1 !       that the node file is already open.
;   655          0648  1 !
;   656          0649  1 ! IMPLICIT OUTPUTS:
;   657          0650  1 !
;   658          0651  1 !       NONE
;   659          0652  1 !
;   660          0653  1 ! ROUTINE VALUE:
;   661          0654  1 ! COMPLETION CODES:
;   662          0655  1 !
;   663          0656  1 !       If the executor node name is found then success (NML$_STS_SUC) is
;   664          0657  1 !       returned.  If the node name is not found a zero length counted string
;   665          0658  1 !       is returned along with failure (NML$_STS_PTY).
;   666          0659  1 !
;   667          0660  1 ! SIDE EFFECTS:
;   668          0661  1 !
;   669          0662  1 !       Destroys contents of PRMBUFFER.
;   670          0663  1 !
;   671          0664  1 !--
;   672          0665  1
;   673          0666  2 BEGIN
;   674          0667  2
;   675          0668  2 MAP
;   676          0669  2     addr : WORD,
;   677          0670  2     bufdsc : REF DESCRIPTOR,
;   678          0671  2     nml$gb_options : BBLOCK [1];
;   679          0672  2
;   680          0673  2 LOCAL
;   681          0674  2     addrdsc : VECTOR [2],
;   682          0675  2     nameptr,
;   683          0676  2     namesize : WORD,
;   684          0677  2     temp,
;   685          0678  2     recdsc : VECTOR [2];
;   686          0679  2
;   687          0680  2 IF .nml$gb_options [nma$v_opt_per] THEN
;   688          0681  3     BEGIN
;   689          0682  3     !
```

```
;  690          0683  3     ! If the node permanent data base file isn't already open, open it.
;  691          0684  3     !
;  692          0685  3     nml$openfile (nma$c_opn_node, nma$c_opn_ac_ro);
;  693          0686  3     recdsc [1] = .prmdsc [1];
;  694          0687  3     addrdsc [0] = 2;
;  695          0688  3     addrdsc [1] = addr;
;  696          0689  3     IF nml$readrecord (nma$c_opn_node,          ! Node perm database file ID
;  697          0690  3                        UPLIT (nma$c_pcno_add),  ! Use ISAM key = node address
;  698          0691  3                        addrdsc,                 ! ISAM key value = node address
;  699          0692  3                        prmdsc,                  ! Read buffer descriptor
;  700          0693  3                        recdsc,                  ! Return data descriptor
;  701          0694  3                        temp) THEN               ! Not used
;  702          0695  4         BEGIN
;  703          0696  4         namesize = 0;
;  704          0697  4         nameptr = 0;
;  705          0698  4         IF nma$searchfld (      recdsc,
;  706          0699  4                                 nma$c_pcno_nna,
;  707          0700  4                                 namesize,
;  708          0701  4                                 nameptr) THEN
;  709          0702  5             BEGIN
;  710          0703  5             CH$MOVE (.namesize, CH$PTR (.nameptr),
;  711          0704  5                              CH$PTR (.bufdsc [dsc$a_pointer]));
;  712          0705  5             .reslen = .namesize;
;  713          0706  5
;  714          0707  5             RETURN nml$_sts_suc
;  715          0708  4             END;
;  716          0709  4         END
;  717          0710  3     END
;  718          0711  2 ELSE
;  719          0712  2     RETURN nml$getvolndnam (.addr, .bufdsc, .reslen);
;  720          0713  2
;  721          0714  2 !
;  722          0715  2 ! No node name found.
;  723          0716  2 !
;  724          0717  2 .reslen = 0;
;  725          0718  2
;  726          0719  2 RETURN nml$_sts_pty
;  727          0720  2
;  728          0721  1 END;                            ! End of NML$GETNODNAM


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                                        000001F6  00028 P.AAG:  .LONG   502                                     ;


                                                        .PSECT  $CODE$,NOWRT,2

                                            007C 00000          .ENTRY  NML$GETNODNAM, Save R2,R3,R4,R5,R6      ; 0627
                                    56 00000000'  00    9E 00002        MOVAB   PRMDSC+4, R6                    ;
                                    5E            1C    C2 00009        SUBL2   #28, SP                         ;
                                       00000000G  00    95 0000C        TSTB    NML$GB_OPTIONS                  ; 0680
                                                  62    18 00012        BGEQ    1$                              ;
                                                  7E    7C 00014        CLRQ    -(SP)                           ; 0685
                                00000000G   00            02    FB 00016        CALLS   #2, NML$OPENFILE        ;
```

```
                         10   AE              66   D0  0001D              MOVL    PRMDSC+4, RECDSC+4                      ; 0686
                         14   AE              02   D0  00021              MOVL    #2, ADDRDSC                             ; 0687
                         18   AE        04    AC   9E  00025              MOVAB   ADDR, ADDRDSC+4                         ; 0688
                                              5E   DD  0002A              PUSHL   SP                                      ; 0689
                                        10    AE   9F  0002C              PUSHAB  RECDSC
                                        FC    A6   9F  0002F              PUSHAB  PRMDSC
                                        20    AE   9F  00032              PUSHAB  ADDRDSC
                                        1C    A6   9F  00035              PUSHAB  P.AAG                                   ; 0690
                                              7E   D4  00038              CLRL    -(SP)                                   ; 0689
                    00000000G   00              06   FB  0003A              CALLS   #6, NML$READRECORD
                                42              50   E9  00041              BLBC    R0, 2$
                                        08    AE   B4  00044              CLRW    NAMESIZE                                ; 0696
                                        04    AE   D4  00047              CLRL    NAMEPTR                                 ; 0697
                                        04    AE   9F  0004A              PUSHAB  NAMEPTR                                 ; 0698
                                        0C    AE   9F  0004D              PUSHAB  NAMESIZE
                                7E      01F4  8F   3C  00050              MOVZWL  #500, -(SP)
                                        18    AE   9F  00055              PUSHAB  RECDSC
                    00000000G   00              04   FB  00058              CALLS   #4, NMA$SEARCHFLD
                                24              50   E9  0005F              BLBC    R0, 2$
                                50        08    AC   D0  00062              MOVL    BUFDSC, R0                              ; 0704
                04  B0          04   BE        08   AE  28  00066              MOVC3   NAMESIZE, @NAMEPTR, @4(R0)
                                 0C   BC        08   AE  3C  0006D              MOVZWL  NAMESIZE, @RESLEN                       ; 0705
                                50              01   D0  00072              MOVL    #1, R0                                  ; 0707
                                                     04  00075              RET
                                7E        08    AC   7D  00076  1$:          MOVQ    BUFDSC, -(SP)                           ; 0712
                                7E        04    AC   3C  0007A              MOVZWL  ADDR, -(SP)
                    00000000V   00              03   FB  0007E              CALLS   #3, NML$GETVOLNDNAM
                                                     04  00085              RET
                                          0C    BC   D4  00086  2$:          CLRL    @RESLEN                                 ; 0717
                                50              0C   CE  00089              MNEGL   #12, R0                                 ; 0719
                                                     04  0008C              RET                                             ; 0721

; Routine Size:  141 bytes,    Routine Base:  $CODE$ + 02E2
```

```
;   730     0722  1 %SBTTL 'NML$GETVOLNDNAM  Get node name given the address'
;   731     0723  1 GLOBAL ROUTINE NML$GETVOLNDNAM (ADDR, BUFDSC, RESLEN) =
;   732     0724  1
;   733     0725  1 !++
;   734     0726  1 ! FUNCTIONAL DESCRIPTION:
;   735     0727  1 !
;   736     0728  1 !       This routine returns the volatile node name that matches the
;   737     0729  1 !       specified address.
;   738     0730  1 !
;   739     0731  1 ! FORMAL PARAMETERS:
;   740     0732  1 !
;   741     0733  1 !       ADDR            Node address.
;   742     0734  1 !       BUFDSC          Address of descriptor of buffer to contain ASCII
;   743     0735  1 !                       node name.
;   744     0736  1 !       RESLEN          Address of longword to contain resulting length
;   745     0737  1 !                       of node name string.
;   746     0738  1 !
;   747     0739  1 ! IMPLICIT INPUTS:
;   748     0740  1 !
;   749     0741  1 !       NONE
;   750     0742  1 !
;   751     0743  1 ! IMPLICIT OUTPUTS:
;   752     0744  1 !
;   753     0745  1 !       NONE
;   754     0746  1 !
;   755     0747  1 ! ROUTINE VALUE:
;   756     0748  1 ! COMPLETION CODES:
;   757     0749  1 !
;   758     0750  1 !       If the node name is found then success (NML$_STS_SUC) is
;   759     0751  1 !       returned.  If the node name is not found a zero length counted string
;   760     0752  1 !       is returned along with failure (NML$_STS_PTY).
;   761     0753  1 !
;   762     0754  1 ! SIDE EFFECTS:
;   763     0755  1 !
;   764     0756  1 !       Destroys contents of PRMBUFFER.
;   765     0757  1 !
;   766     0758  1 !--
;   767     0759  1
;   768     0760  2 BEGIN
;   769     0761  2
;   770     0762  2 MAP
;   771     0763  2     addr    : WORD,
;   772     0764  2     bufdsc : REF DESCRIPTOR;
;   773     0765  2
;   774   P 0766  2 $nfbdsc(nfbdsc, show, , ndi
;   775   P 0767  2                   ,tad,           ! Search key 1 = Transformed Address, oper1 = eql
;   776   P 0768  2                   ,nfb$c_wildcard,! Search key 2 = wildcard, oper2 = eql
;   777     0769  2                   ,nna);
;   778     0770  2
;   779     0771  2 LOCAL
;   780     0772  2     p2dsc : VECTOR [2],
;   781     0773  2     nameptr,
;   782     0774  2     namesize : WORD,
;   783     0775  2     node_addr;
;   784     0776  2
;   785     0777  2 node_addr = .addr;
;   786     0778  2
```

```
;  787          0779  2 IF .addr EQL 0 THEN                          ! If zero address then
;  788          0780  2     nml$getexeadr (node_addr);  ! get the real executor address
;  789          0781  2 nml$bldp2(0, .node_addr, -1, 0, p2bfdsc, p2dsc);
;  790          0782  2
;  791          0783  2 IF nml$netqio ( nfbdsc,
;  792          0784  2                  p2dsc,
;  793          0785  2                  namesize,
;  794          0786  2                  prmdsc) THEN
;  795          0787  3     BEGIN
;  796          0788  3     nameptr = .prmdsc [1];
;  797          0789  3     namesize = .(.nameptr)<0,16>;
;  798          0790  3     CH$MOVE (.namesize, CH$PTR (.nameptr,2), .bufdsc [dsc$a_pointer]);
;  799          0791  3     .reslen = .namesize;
;  800          0792  3     RETURN nml$_sts_suc
;  801          0793  2     END;
;  802          0794  2 !
;  803          0795  2 ! No node name found.
;  804          0796  2 !
;  805          0797  2 .reslen = 0;
;  806          0798  2
;  807          0799  2 RETURN nml$_sts_pty
;  808          0800  2
;  809          0801  1 END;                          ! End of NML$GETVOLNDNAM


                                                      .PSECT  $PLIT$,NOWRT,NOEXE,2

                                            0000001C  0002C P.AAH:  .LONG   28                                      ;
                                            00000000' 00030         .ADDRESS U.3                                    ;

                                                      .PSECT  $OWN$,NOEXE,2

                                                 22   001CC ;_NFB
                                                       U.3:       .BYTE   34                                        ;
                                                 00   001CD       .BYTE   0                                         ;
                                                 02   001CE       .BYTE   2                                         ;
                                                 00   001CF       .BYTE   0                                         ;
                                            02010010  001D0       .LONG   33619984                                  ;
                                            00000001  001D4       .LONG   1                                         ;
                                                 00   001D8       .BYTE   0                                         ;
                                                 00   001D9       .BYTE   0                                         ;
                                               0000   001DA       .WORD   0                                         ;
                                            02020043  001DC       .LONG   33685571                                  ;
                                            00000000  001E0       .LONG   0                                         ;
                                                      001E4       .BLKB   4

                                                       U.4=                P.AAH

                                                      .PSECT  $CODE$,NOWRT,2

                                                      007C 00000    .ENTRY  NML$GETVOLNDNAM, Save R2,R3,R4,R5,R6      ; 0723
                                        56 00000000'  00   9E 00002 MOVAB   P2BFDSC, R6                              ;
                                        5E            0C   C2 00009 SUBL2   #12, SP                                 ;
                                        7E         04 AC   3C 0000C MOVZWL  ADDR, NODE_ADDR                         ; 0777
                                                   07   12 00010 BNEQ    1$                                        ; 0779
```

```
                                    5E  DD 00012          PUSHL   SP                                          ; 0780
                          FEDB  CF  01  FB 00014          CALLS   #1, NML$GETEXEADR                           ;
                                08  AE  9F 00019 1$:      PUSHAB  P2DSC                                       ; 0781
                                    56  DD 0001C          PUSHL   R6                                          ;
                                    7E  D4 0001E          CLRL    -(SP)                                       ;
                                7E  01  CE 00020          MNEGL   #1, -(SP)                                   ;
                                10  AE  DD 00023          PUSHL   NODE_ADDR                                   ;
                                    7E  D4 00026          CLRL    -(SP)                                       ;
                          FC64  CF  06  FB 00028          CALLS   #6, NML$BLDP2                               ;
                                08  A6  9F 0002D          PUSHAB  PRMDSC                                      ; 0783
                                08  AE  9F 00030          PUSHAB  NAMESIZE                                    ;
                                10  AE  9F 00033          PUSHAB  P2DSC                                       ;
                                2C  A6  9F 00036          PUSHAB  NFBDSC                                      ;
                00000000G       00  FB 00039          CALLS   #4, NML$NETQIO                              ;
                                1C  50  E9 00040          BLBC    R0, 2$                                      ;
                                51  0C  A6  D0 00043      MOVL    PRMDSC+4, NAMEPTR                           ; 0788
                            04  AE  61  B0 00047          MOVW    (NAMEPTR), NAMESIZE                         ; 0789
                                50  08  AC  D0 0004B      MOVL    BUFDSC, R0                                  ; 0790
          04    B0          02  A1  04  AE  28 0004F      MOVC3   NAMESIZE, 2(NAMEPTR), @4(R0)                ;
                            0C  BC  04  AE  3C 00056      MOVZWL  NAMESIZE, @RESLEN                           ; 0791
                                50      01  D0 0005B      MOVL    #1, R0                                      ; 0792
                                        04  0005E          RET                                                 ;
                                0C  BC  D4 0005F 2$:      CLRL    @RESLEN                                     ; 0797
                                50  0C  CE 00062          MNEGL   #12, R0                                     ; 0799
                                        04  00065          RET                                                 ; 0801

; Routine Size:  102 bytes,    Routine Base:  $CODE$ + 036F
```

```
;  811          0802  1 %SBTTL 'NML$GETNODADR  Get node address given the name'
;  812          0803  1 GLOBAL ROUTINE NML$GETNODADR (NAMLEN, NAMPTR, ADDR) =
;  813          0804  1
;  814          0805  1 !++
;  815          0806  1 ! FUNCTIONAL DESCRIPTION:
;  816          0807  1 !
;  817          0808  1 !       This routine returns the node address that matches the
;  818          0809  1 !       specified name.
;  819          0810  1 !
;  820          0811  1 ! FORMAL PARAMETERS:
;  821          0812  1 !
;  822          0813  1 !       ADDR            Address of word to contain node address.
;  823          0814  1 !
;  824          0815  1 ! IMPLICIT INPUTS:
;  825          0816  1 !
;  826          0817  1 !       NML$GB_OPTIONS contains the command message options.
;  827          0818  1 !
;  828          0819  1 !       If this is a permanent data base operation then it is assumed
;  829          0820  1 !       that the node file is already open.
;  830          0821  1 !
;  831          0822  1 ! IMPLICIT OUTPUTS:
;  832          0823  1 !
;  833          0824  1 !       NONE
;  834          0825  1 !
;  835          0826  1 ! ROUTINE VALUE:
;  836          0827  1 ! COMPLETION CODES:
;  837          0828  1 !
;  838          0829  1 !       If the node address is found then success (NML$_STS_SUC) is
;  839          0830  1 !       returned.  If the node address is not found a zero address
;  840          0831  1 !       is returned along with failure (NML$_STS_PTY).
;  841          0832  1 !
;  842          0833  1 ! SIDE EFFECTS:
;  843          0834  1 !
;  844          0835  1 !       Destroys contents of PRMBUFFER.
;  845          0836  1 !
;  846          0837  1 !--
;  847          0838  1
;  848          0839  2 BEGIN
;  849          0840  2
;  850          0841  2 MAP
;  851          0842  2     nml$gb_options : BBLOCK [1];
;  852          0843  2
;  853        P 0844  2 $nfbdsc(nfbdsc, show, , ndi
;  854        P 0845  2                 ,nna,             ! Search key 1 = Node name, oper1 = eql
;  855        P 0846  2                 ,nfb$c_wildcard,! Search key 2 = wildcard, oper2 = eql
;  856          0847  2                 ,tad);
;  857          0848  2
;  858          0849  2 LOCAL
;  859          0850  2     fldadr,
;  860          0851  2     fldsize,
;  861          0852  2     p2dsc     : VECTOR [2],
;  862          0853  2     ptr,
;  863          0854  2     key       : WORD,
;  864          0855  2     temp,
;  865          0856  2     recdsc    : VECTOR [2];
;  866          0857  2
;  867          0858  2 IF .nml$gb_options [nma$v_opt_per] THEN
```

```
;  868     0859  3     BEGIN
;  869     0860  3     !
;  870     0861  3     ! If the node permanent data base file isn't already open, open it.
;  871     0862  3     !
;  872     0863  3     nml$openfile (nma$c_opn_node, nma$c_opn_ac_ro);
;  873     0864  3     recdsc [1] = .prmdsc [1];
;  874     0865  3     IF nml$readrecord (nma$c_opn_node,          ! Node perm database ID
;  875     0866  3                        UPLIT (nma$c_pcno_nna),  ! Use ISAM key = node name
;  876     0867  3                        namlen,                  ! ISAM key value dsc = node name
;  877     0868  3                        prmdsc,                  ! Read buffer descriptor
;  878     0869  3                        recdsc,                  ! Return data descriptor
;  879     0870  3                        temp) THEN               ! Not used
;  880     0871  4         BEGIN
;  881     0872  4         fldadr = 0;
;  882     0873  4         IF nma$searchfld (      recdsc,
;  883     0874  4                                 nma$c_pcno_add,
;  884     0875  4                                 fldsize,
;  885     0876  4                                 fldadr) THEN
;  886     0877  5             BEGIN
;  887     0878  5             CH$MOVE (2, .fldadr, .addr);
;  888     0879  5             RETURN nml$_sts_suc
;  889     0880  4             END;
;  890     0881  4         END
;  891     0882  3     END
;  892     0883  2 ELSE
;  893     0884  2     IF nml$getvolndadr (.namlen, .namptr, .addr) THEN
;  894     0885  2         RETURN nml$_sts_suc;
;  895     0886  2 !
;  896     0887  2 ! No node address found.
;  897     0888  2 !
;  898     0889  2 (.addr)<0,16> = 0;
;  899     0890  2
;  900     0891  2 RETURN nml$_sts_pty
;  901     0892  2
;  902     0893  1 END;                                ! End of NML$GETNODADR


                                .PSECT  $PLIT$,NOWRT,NOEXE,2

0000001C  00034 P.AAI:  .LONG   28                                          ;
00000000' 00038         .ADDRESS U.5                                        ;
000001F4  0003C P.AAJ:  .LONG   500                                         ;

                                .PSECT  $OWN$,NOEXE,2

      22  001E8 ;_NFB
                U.5:    .BYTE   34                                          ;
      00  001E9         .BYTE   0                                           ;
      02  001EA         .BYTE   2                                           ;
      00  001EB         .BYTE   0                                           ;
02020043  001EC         .LONG   33685571                                    ;
00000001  001F0         .LONG   1                                           ;
      00  001F4         .BYTE   0                                           ;
      00  001F5         .BYTE   0                                           ;
    0000  001F6         .WORD   0                                           ;
02010010  001F8         .LONG   33619984                                    ;
```

```
                                00000000  001FC          .LONG   0                                               ;
                                          00200          .BLKB   4

                                                U.6=                    P.AAI

                                                       .PSECT  $CODE$,NOWRT,2

                                     0004 00000        .ENTRY  NML$GETNODADR, Save R2                          ; 0803
                  52 00000000'   00   9E 00002        MOVAB   PRMDSC+4, R2                                    ;
                  5E             1C   C2 00009        SUBL2   #28, SP                                         ;
                     00000000G   00   95 0000C        TSTB    NML$GB_OPTIONS                                  ; 0858
                                 49   18 00012        BGEQ    1$                                              ;
                                 7E   7C 00014        CLRQ    -(SP)                                           ; 0863
00000000G  00                    02   FB 00016        CALLS   #2, NML$OPENFILE                                ;
        10 AE                    62   D0 0001D        MOVL    PRMDSC+4, RECDSC+4                              ; 0864
                                 5E   DD 00021        PUSHL   SP                                              ; 0865
                           10    AE   9F 00023        PUSHAB  RECDSC                                          ;
                           FC    A2   9F 00026        PUSHAB  PRMDSC                                          ;
                           04    AC   9F 00029        PUSHAB  NAMLEN                                          ;
                           30    A2   9F 0002C        PUSHAB  P.AAJ                                           ; 0866
                                 7E   D4 0002F        CLRL    -(SP)                                           ; 0865
00000000G  00                    06   FB 00031        CALLS   #6, NML$READRECORD                              ;
        37                       50   E9 00038        BLBC    R0, 3$                                          ;
                           04    AE   D4 0003B        CLRL    FLDADR                                          ; 0872
                           04    AE   9F 0003E        PUSHAB  FLDADR                                          ; 0873
                           0C    AE   9F 00041        PUSHAB  FLDSIZE                                         ;
                  7E     01F6   8F   3C 00044        MOVZWL  #502, -(SP)                                     ;
                           18    AE   9F 00049        PUSHAB  RECDSC                                          ;
00000000G  00                    04   FB 0004C        CALLS   #4, NMA$SEARCHFLD                               ;
        1C                       50   E9 00053        BLBC    R0, 3$                                          ;
        0C BC            04      BE   B0 00056        MOVW    @FLDADR, @ADDR                                  ; 0878
                                 11   11 0005B        BRB     2$                                              ; 0879
                  7E     08      AC   7D 0005D  1$:    MOVQ    NAMPTR, -(SP)                                   ; 0884
                           04    AC   DD 00061        PUSHL   NAMLEN                                          ;
00000000V  00                    03   FB 00064        CALLS   #3, NML$GETVOLNDADR                             ;
        04                       50   E9 0006B        BLBC    R0, 3$                                          ;
        50                       01   D0 0006E  2$:    MOVL    #1, R0                                          ; 0885
                                 04   00071        RET                                                     ;
                           0C    BC   B4 00072  3$:    CLRW    @ADDR                                           ; 0889
                  50             0C   CE 00075        MNEGL   #12, R0                                         ; 0891
                                 04   00078        RET                                                     ; 0893

; Routine Size:  121 bytes,    Routine Base:  $CODE$ + 03D5
```

```
;  904          0894  1 %SBTTL 'NML$GETVOLNDADR  Get volatile node address given the name'
;  905          0895  1 GLOBAL ROUTINE NML$GETVOLNDADR (NAMLEN, NAMPTR, ADDR) =
;  906          0896  1
;  907          0897  1 !++
;  908          0898  1 ! FUNCTIONAL DESCRIPTION:
;  909          0899  1 !
;  910          0900  1 !       This routine returns the node address from the volatile data base
;  911          0901  1 !       that matches the specified name.
;  912          0902  1 !
;  913          0903  1 ! FORMAL PARAMETERS:
;  914          0904  1 !
;  915          0905  1 !       ADDR            Address of word to contain node address.
;  916          0906  1 !
;  917          0907  1 ! IMPLICIT INPUTS:
;  918          0908  1 !
;  919          0909  1 !       NONE
;  920          0910  1 !
;  921          0911  1 ! IMPLICIT OUTPUTS:
;  922          0912  1 !
;  923          0913  1 !       NONE
;  924          0914  1 !
;  925          0915  1 ! ROUTINE VALUE:
;  926          0916  1 ! COMPLETION CODES:
;  927          0917  1 !
;  928          0918  1 !       If the node address is found then success (NML$_STS_SUC) is
;  929          0919  1 !       returned.  If the node address is not found a zero address
;  930          0920  1 !       is returned along with failure (NML$_STS_PTY).
;  931          0921  1 !
;  932          0922  1 ! SIDE EFFECTS:
;  933          0923  1 !
;  934          0924  1 !       Destroys contents of PRMBUFFER.
;  935          0925  1 !
;  936          0926  1 !--
;  937          0927  1
;  938          0928  2 BEGIN
;  939          0929  2
;  940        P 0930  2 $nfbdsc(nfbdsc, show, , ndi
;  941        P 0931  2                   ,nna,           ! Search key 1 = node name, oper1 = eql
;  942        P 0932  2                   ,nfb$c_wildcard,! Search key 2 = wildcard, oper2 = eql
;  943          0933  2                   ,tad);
;  944          0934  2
;  945          0935  2 LOCAL
;  946          0936  2     p2dsc     : VECTOR [2],
;  947          0937  2     ptr;
;  948          0938  2
;  949          0939  2 nml$bldp2(.namlen, .namptr, -1, 0, p2bfdsc, p2dsc);
;  950          0940  2 IF nml$netqio ( nfbdsc,
;  951          0941  2                   p2dsc,
;  952          0942  2                   0,
;  953          0943  2                   prmdsc) THEN
;  954          0944  3     BEGIN
;  955          0945  3     MAP
;  956          0946  3         ptr: REF BBLOCK,
;  957          0947  3         nml$gw_vol_exec_addr: BBLOCK;
;  958          0948  3
;  959          0949  3     ptr = .prmdsc [1];
;  960          0950  3     IF CH$RCHAR (nml$gb_ncp_version) LEQ 3 THEN
```

```
;  961          0951  4          BEGIN
;  962          0952  4          IF .ptr [nma$v_area] EQL .nml$gw_vol_exec_addr [nma$v_area] THEN
;  963          0953  4              ptr [nma$v_area] = 0;
;  964          0954  3          END;
;  965          0955  3      CH$MOVE (2, .ptr, .addr);
;  966          0956  3      RETURN nml$_sts_suc
;  967          0957  2      END;
;  968          0958  2
;  969          0959  2  !
;  970          0960  2  ! No node address found.
;  971          0961  2  !
;  972          0962  2  (.addr)<0,16> = 0;
;  973          0963  2  RETURN nml$_sts_pty
;  974          0964  2
;  975          0965  1  END;                                ! End of NML$GETNODADR


                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                    0000001C  00040 P.AAK:  .LONG   28                                      ;
                                    00000000' 00044         .ADDRESS U.7                                    ;

                                                .PSECT  $OWN$,NOEXE,2

                                          22  00204 ;_NFB
                                                      U.7:  .BYTE   34                                      ;
                                          00  00205         .BYTE   0                                       ;
                                          02  00206         .BYTE   2                                       ;
                                          00  00207         .BYTE   0                                       ;
                                    02020043  00208         .LONG   33685571                                ;
                                    00000001  0020C         .LONG   1                                       ;
                                          00  00210         .BYTE   0                                       ;
                                          00  00211         .BYTE   0                                       ;
                                        0000  00212         .WORD   0                                       ;
                                    02010010  00214         .LONG   33619984                                ;
                                    00000000  00218         .LONG   0                                       ;
                                              0021C         .BLKB   4

                                                U.8=                P.AAK

                                                .PSECT  $CODE$,NOWRT,2

                                              0004 00000        .ENTRY  NML$GETVOLNDADR, Save R2             ; 0895
                                          52 00000000'  00  9E  00002   MOVAB   P2BFDSC, R2                   ;
                                          5E            08  C2  00009   SUBL2   #8, SP                       ;
                                                  4004  8F  BB  0000C   PUSHR   #^M<R2,SP>                   ; 0939
                                                        7E  D4  00010   CLRL    -(SP)                       ;
                                          7E            01  CE  00012   MNEGL   #1, -(SP)                    ;
                                          7E        04  AC  7D  00015   MOVQ    NAMLEN, -(SP)               ;
                                      FB94  CF            06  FB  00019   CALLS   #6, NML$BLDP2              ;
                                                    08  A2  9F  0001E   PUSHAB  PRMDSC                      ; 0940
                                                        7E  D4  00021   CLRL    -(SP)                       ;
                                                    08  AE  9F  00023   PUSHAB  P2DSC                       ;
                                                    40  A2  9F  00026   PUSHAB  NFBDSC                      ;
                                    00000000G     00        04  FB  00029   CALLS   #4, NML$NETQIO          ;
```

```
                                 2A           50  E9 00030          BLBC    R0, 2$                                                  ;
                                 50       0C  A2  D0 00033          MOVL    PRMDSC+4, PTR                                           ; 0949
                                 03 00000000G 00  91 00037          CMPB    NML$GB_NCP_VERSION, #3                                  ; 0950
                                              15  1A 0003E          BGTRU   1$                                                      ;
          51 00000000'  00                    06  EF 00040          EXTZV   #2, #6, NML$GW_VOL_EXEC_ADDR+1, R1                      ; 0952
          51            60                    06  ED 00049          CMPZV   #10, #6, (PTR), R1                                      ;
                                              05  12 0004E          BNEQ    1$                                                      ;
                                 01   A0   FC  8F  8A 00050          BICB2   #252, 1(PTR)                                            ; 0953
                                 0C   BC       60  B0 00055 1$:     MOVW    (PTR), @ADDR                                            ; 0955
                                 50            01  D0 00059          MOVL    #1, R0                                                  ; 0956
                                               04  0005C          RET                                                             ;
                                          0C   BC  B4 0005D 2$:     CLRW    @ADDR                                                   ; 0962
                                 50            0C  CE 00060          MNEGL   #12, R0                                                 ; 0963
                                               04  00063          RET                                                             ; 0965

; Routine Size:  100 bytes,    Routine Base:  $CODE$ + 044E
```

```
;  977     0966  1 %SBTTL 'NML$GETEXEID  Get executor node id'
;  978     0967  1 GLOBAL ROUTINE NML$GETEXEID (BUFDSC, RESLEN) =
;  979     0968  1
;  980     0969  1 !++
;  981     0970  1 ! FUNCTIONAL DESCRIPTION:
;  982     0971  1 !
;  983     0972  1 !       This routine returns the executor node address followed by
;  984     0973  1 !       the node name.
;  985     0974  1 !
;  986     0975  1 ! FORMAL PARAMETERS:
;  987     0976  1 !
;  988     0977  1 !       BUFDSC          Address of descriptor of buffer to contain ASCII
;  989     0978  1 !                       node name.
;  990     0979  1 !       RESLEN          Resulting length of node name string.
;  991     0980  1 !
;  992     0981  1 ! IMPLICIT INPUTS:
;  993     0982  1 !
;  994     0983  1 !       If this is a permanent data base operation then it is assumed
;  995     0984  1 !       that the executor and node files are already open.
;  996     0985  1 !
;  997     0986  1 ! IMPLICIT OUTPUTS:
;  998     0987  1 !
;  999     0988  1 !       NONE
; 1000     0989  1 !
; 1001     0990  1 ! ROUTINE VALUE:
; 1002     0991  1 ! COMPLETION CODES:
; 1003     0992  1 !
; 1004     0993  1 !       If the executor node name is found then success (NML$_STS_SUC) is
; 1005     0994  1 !       returned.  If the node name is not found a zero length counted string
; 1006     0995  1 !       is returned along with failure (NML$_STS_PTY).
; 1007     0996  1 !
; 1008     0997  1 ! SIDE EFFECTS:
; 1009     0998  1 !
; 1010     0999  1 !       NONE
; 1011     1000  1 !
; 1012     1001  1 !--
; 1013     1002  1
; 1014     1003  2 BEGIN
; 1015     1004  2
; 1016     1005  2 MAP
; 1017     1006  2     bufdsc          : REF DESCRIPTOR;
; 1018     1007  2
; 1019     1008  2 LOCAL
; 1020     1009  2     addr : WORD,
; 1021     1010  2     nambuf : VECTOR [6, BYTE],
; 1022     1011  2     namdsc : VECTOR [2],
; 1023     1012  2     namlen,
; 1024     1013  2     ptr;
; 1025     1014  2
; 1026     1015  2 ptr = ch$ptr (.bufdsc [dsc$a_pointer]);
; 1027     1016  2
; 1028     1017  2 nml$getexeadr (addr);           ! Get address
; 1029     1018  2
; 1030     1019  2 namdsc [0] = 6;
; 1031     1020  2 namdsc [1] = nambuf;
; 1032     1021  2
; 1033     1022  2 nml$getexenam (namdsc, namlen); ! Get name
```

```
;  1034          1023  2
;  1035          1024  2 ch$wchar_a (.(addr)<0,8>, ptr);
;  1036          1025  2 ch$wchar_a (.(addr)<8,8>, ptr);
;  1037          1026  2 CH$WCHAR_A (.namlen OR nma$m_ent_exe, ptr);
;  1038          1027  2 ptr = CH$MOVE (.namlen, .namdsc [1], .ptr);
;  1039          1028  2 .reslen = .ptr - .bufdsc [dsc$a_pointer];
;  1040          1029  2
;  1041          1030  2 RETURN nml$_sts_suc
;  1042          1031  2
;  1043          1032  1 END;                                   ! End of NML$GETEXEID


                                                  0067C 00000        .ENTRY  NML$GETEXEID, Save R2,R3,R4,R5,R6                ; 0967
                                        5E              18   C2 00002  SUBL2   #24, SP                                        ;
                                        56        04    AC   D0 00005  MOVL    BUFDSC, R6                                    ; 1015
                                        53        04    A6   D0 00009  MOVL    4(R6), PTR                                    ;
                                                        5E   DD 0000D  PUSHL   SP                                            ; 1017
                                  FD9D  CF              01   FB 0000F  CALLS   #1, NML$GETEXEADR                             ;
                                    08  AE              06   D0 00014  MOVL    #6, NAMDSC                                    ; 1019
                                    0C  AE        10    AE   9E 00018  MOVAB   NAMBUF, NAMDSC+4                              ; 1020
                                                  04    AE   9F 0001D  PUSHAB  NAMLEN                                        ; 1022
                                                  0C    AE   9F 00020  PUSHAB  NAMDSC                                        ;
                                  FDBE  CF              02   FB 00023  CALLS   #2, NML$GETEXENAM                             ;
                                        83              6E   B0 00028  MOVW    ADDR, (PTR)+                                  ; 1024
                              83        04    AE        80   8F 89 0002B  BISB3   #128, NAMLEN, (PTR)+                       ; 1026
                              63        0C    BE  04    AE   28 00031  MOVC3   NAMLEN, @NAMDSC+4, (PTR)                     ; 1027
                        08    BC              53  04    A6   C3 00037  SUBL3   4(R6), PTR, @RESLEN                          ; 1028
                                        50              01   D0 0003D  MOVL    #1, R0                                        ; 1030
                                                        04   00040  RET                                                     ; 1032

; Routine Size:  65 bytes,    Routine Base:  $CODE$ + 04B2
```

```
;  1045          1033  1 %SBTTL 'NML$GETINFTABS  Get NFB and information table'
;  1046          1034  1 GLOBAL ROUTINE NML$GETINFTABS (ENT, INF, NFBDSC, TABDSC, COPY) =
;  1047          1035  1
;  1048          1036  1 !++
;  1049          1037  1 ! FUNCTIONAL DESCRIPTION:
;  1050          1038  1 !
;  1051          1039  1 !       This routine returns the NFB descriptor address and the corresponding
;  1052          1040  1 !       table address based on the internal entity type and the information
;  1053          1041  1 !       type specified in the NCP SHOW command message.
;  1054          1042  1 !
;  1055          1043  1 ! FORMAL PARAMETERS:
;  1056          1044  1 !
;  1057          1045  1 !       ENT             Internal entity type code.
;  1058          1046  1 !       INF             Internal information type code.
;  1059          1047  1 !       NFBDSC          Address of longword to get NFB descriptor address.
;  1060          1048  1 !       TABDSC          Address of longword to get table descriptor address.
;  1061          1049  1 !       COPY            (Temporary parameter) If 1, copy the NFB to the
;  1062          1050  1 !                       buffer specified by NFBDSC, and fill in NFBDSC length.
;  1063          1051  1 !
;  1064          1052  1 ! ROUTINE VALUE:
;  1065          1053  1 ! COMPLETION CODES:
;  1066          1054  1 !
;  1067          1055  1 !       If the descriptors are found for the specified entity and information
;  1068          1056  1 !       type then success (NML$_STS_SUC) is returned.  If the information type
;  1069          1057  1 !       is invalid for the entity then an error message (NML$_STS_FUN) is
;  1070          1058  1 !       signalled.
;  1071          1059  1 !
;  1072          1060  1 ! SIDE EFFECTS:
;  1073          1061  1 !
;  1074          1062  1 !       NONE
;  1075          1063  1 !
;  1076          1064  1 !--
;  1077          1065  1
;  1078          1066  2 BEGIN
;  1079          1067  2
;  1080          1068  2 LOCAL
;  1081          1069  2         single_ent_nfbdsc : REF DESCRIPTOR,
;  1082          1070  2         enttab : REF BBLOCKVECTOR [, 8]; ! Address of entity table
;  1083          1071  2
;  1084          1072  2 entlab = .nml$al_entinftab [.ent];
;  1085          1073  2 !
;  1086          1074  2 ! Return address of table used to format the NICE response message for
;  1087          1075  2 ! this entity.
;  1088          1076  2 !
;  1089          1077  2 .tabdsc = .enttab [.inf, 4,0,32,0];
;  1090          1078  2
;  1091          1079  2 !
;  1092          1080  2 ! Return the canned NFB and NFB descriptor for getting the SHOW info
;  1093          1081  2 ! from NETACP.
;  1094          1082  2 !
;  1095          1083  2 IF NOT .copy THEN
;  1096          1084  3         BEGIN
;  1097          1085  3         .nfbdsc = .enttab [.inf, 0,0,32,0];
;  1098          1086  3
;  1099          1087  3         IF ..nfbdsc EQLA 0
;  1100          1088  3         THEN
;  1101          1089  3                 nml$error_1 (nma$c_sts_fun);
```

```
;  1102          1090  3          END
;  1103          1091  2 ELSE
;  1104          1092  2          !
;  1105          1093  2          ! For the new QIO interface, each plural entity show should
;  1106          1094  2          ! be modified to use the following path.  When I get around to it.
;  1107          1095  2          ! For plural entity SHOWs, copy the single entity NFB to local
;  1108          1096  2          ! storage in the calling routine, where it will be modified to
;  1109          1097  2          ! issue a plural entity SHOW.
;  1110          1098  2          !
;  1111          1099  2          BEGIN
;  1112          1100  2
;  1113          1101  2          MAP
;  1114          1102  2                  nfbdsc : REF DESCRIPTOR;
;  1115          1103  2
;  1116          1104  2          single_ent_nfbdsc = .enttab [.inf, 0,0,32,0];
;  1117          1105  2          IF .single_ent_nfbdsc EQLA 0 THEN
;  1118          1106  2                  nml$error_1 (nma$c_sts_fun);
;  1119          1107  2          .nfbdsc = ..single_ent_nfbdsc;                  ! Set up NFB length.
;  1120          1108  2          CH$MOVE (.single_ent_nfbdsc [dsc$w_length],
;  1121          1109  2                   .single_ent_nfbdsc [dsc$a_pointer],
;  1122          1110  2                   .nfbdsc [dsc$a_pointer]);
;  1123          1111  2          END;
;  1124          1112  2
;  1125          1113  2 RETURN nml$_sts_suc
;  1126          1114  2
;  1127          1115  1 END;                                  ! End of NML$GETINFTABS


                                    007C 00000          .ENTRY  NML$GETINFTABS, Save R2,R3,R4,R5,R6          ; 1034
                              56 00000000G   00   9E 00002  MOVAB   NML$ERROR_1, R6                          ;
                              50       04    AC   D0 00009  MOVL    ENT, R0                                 ; 1072
                              51 00000000G0040   D0 0000D  MOVL    NML$AL_ENTINFTAB[R0], ENTTAB            ;
                              50       08    AC   D0 00015  MOVL    INF, R0                                 ; 1077
                              50          6140   7F 00019  MOVAQ   (ENTTAB)[R0], R0                       ;
                        10    BC       04    A0   D0 0001D  MOVL    4(R0), @TABDSC                          ;
                              53       0C    AC   D0 00022  MOVL    NFBDSC, R3                              ; 1085
                              0D       14    AC   E8 00026  BLBS    COPY, 1$                                ; 1083
                              63             60   D0 0002A  MOVL    (R0), (R3)                              ; 1085
                                             1C   12 0002D  BNEQ    3$                                      ; 1087
                              7E             01   CE 0002F  MNEGL   #1, -(SP)                              ; 1089
                              66             01   FB 00032  CALLS   #1, NML$ERROR_1                         ;
                                             14   11 00035  BRB     3$                                      ; 1083
                              52             60   D0 00037  1$:    MOVL    (R0), SINGLE_ENT_NFBDSC         ; 1104
                                             06   12 0003A  BNEQ    2$                                      ; 1105
                              7E             01   CE 0003C  MNEGL   #1, -(SP)                              ; 1106
                              66             01   FB 0003F  CALLS   #1, NML$ERROR_1                         ;
                              63             62   D0 00042  2$:    MOVL    (SINGLE_ENT_NFBDSC), (R3)       ; 1107
                        04    B3       04    B2   28 00045  MOVC3   (SINGLE_ENT_NFBDSC), -                 ; 1110
                                                                   @4(SINGLE_ENT_NFBDSC), @4(R3)           ;
                              50             01   D0 0004B  3$:    MOVL    #1, R0                          ; 1113
                                                04 0004E  RET                                              ; 1115

; Routine Size:  79 bytes,    Routine Base:  $CODE$ + 04F3
```

```
;  1129          1116  1 %SBTTL 'NML$FIX_NODE_NUM  Fix node address parameter (action routine)'
;  1130          1117  1 GLOBAL ROUTINE NML$FIX_NODE_NUM (NODE_ADDR) =
;  1131          1118  1
;  1132          1119  1 !++
;  1133          1120  1 ! FUNCTIONAL DESCRIPTION:
;  1134          1121  1 !
;  1135          1122  1 !       This is an NPARSE action that checks the node address.  If the area
;  1136          1123  1 !       number is 0 it can be one of two cases:
;  1137          1124  1 !               The NCP is a Phase IV NCP, and user did not specify an area
;  1138          1125  1 !               number in the NCP command.  In this case, assume the user
;  1139          1126  1 !               means area 1 (since 0 is an invalid area number).
;  1140          1127  1 !
;  1141          1128  1 !               the NCP is a Phase III NCP and therefore doesn't understand
;  1142          1129  1 !               area numbers.  In this case, assume the user means the
;  1143          1130  1 !               executor node's area.
;  1144          1131  1 !
;  1145          1132  1 ! FORMAL PARAMETERS:
;  1146          1133  1 !       NODE_ADDR       Address of Node address to fix.
;  1147          1134  1 !
;  1148          1135  1 ! IMPLICIT INPUTS:
;  1149          1136  1 !       None
;  1150          1137  1 !
;  1151          1138  1 ! IMPLICIT OUTPUTS:
;  1152          1139  1 !       None
;  1153          1140  1 !
;  1154          1141  1 !--
;  1155          1142  1
;  1156          1143  2 BEGIN
;  1157          1144  2
;  1158          1145  2 MAP
;  1159          1146  2     node_addr    : REF BBLOCK [2],
;  1160          1147  2     nml$gb_options : BBLOCK [1];
;  1161          1148  2
;  1162          1149  2 LOCAL
;  1163          1150  2     exec_addr    : BBLOCK [2];
;  1164          1151  2
;  1165          1152  2 !
;  1166          1153  2 ! If the node address is 0, then it's the executor, so leave it that way.
;  1167          1154  2 ! If the area number of the address is 0, then change it.
;  1168          1155  2 !
;  1169          1156  2 IF .node_addr [nma$v_addr] NEQ 0 AND
;  1170          1157  2    .node_addr [nma$v_area] EQL 0 THEN
;  1171          1158  3     BEGIN
;  1172          1159  3     !
;  1173          1160  3     ! Get the executor address from the volatile database if the NICE command
;  1174          1161  3     ! is a volatile database command and from the permanent database if the
;  1175          1162  3     ! NICE command is a permanent database command.  Use the executor's area
;  1176          1163  3     ! number for the node address supplied.
;  1177          1164  3     !
;  1178          1165  3     nml$getexeadr (exec_addr);
;  1179          1166  3     node_addr [nma$v_area] = .exec_addr [nma$v_area];
;  1180          1167  2     END;
;  1181          1168  2 RETURN nml$_sts_suc
;  1182          1169  2
;  1183          1170  1 END;                                 ! End of NML$FIX_NODE_NUM
```

```
                                                0004 00000          .ENTRY  NML$FIX_NODE_NUM, Save R2                       ; 1117
                                           5E   04   C2 00002          SUBL2   #4, SP                                          ;
                                           52   04   AC D0 00005          MOVL    NODE_ADDR, R2                              ; 1156
                                      03FF 8F        62 B3 00009          BITW    (R2), #1023                                ;
                                                     19 13 0000E          BEQL    1$                                         ;
                                        FC 8F   01   A2 93 00010          BITB    1(R2), #252                                ; 1157
                                                     12 12 00015          BNEQ    1$                                         ;
                                                     5E DD 00017          PUSHL   SP                                         ; 1165
                                      FD03 CF        01 FB 00019          CALLS   #1, NML$GETEXEADR                          ;
                50      01   AE                    02 EF 0001E          EXTZV   #2, #6, EXEC_ADDR+1, R0                    ; 1166
                62           06                    50 F0 00024          INSV    R0, #10, #6, (R2)                          ;
                                           50        01 D0 00029 1$:     MOVL    #1, R0                                     ; 1168
                                                     04 0002C            RET                                                ; 1170

; Routine Size:  45 bytes,    Routine Base:  $CODE$ + 0542
```

```
; 1185          1171  1 END
; 1186          1172  1
; 1187          1173  0 ELUDOM



                                                        .EXTRN  LIB$SIGNAL

;                               PSECT SUMMARY

;       Name                    Bytes                           Attributes

;  $GLOBAL$                       52  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $OWN$                         544  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                         72  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                       1391  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        35        10          27        00:00.1
;  _$255$DUA28:[SHRLIB]NMALIBRY.L32;1     887        10         1          47        00:00.2
;  _$255$DUA28:[SHRLIB]NET.L32;1         1279        10         0          63        00:00.3
;  _$255$DUA28:[SYSLIB]STARLET.L32;1     9776         2         0         581        00:03.2



;                                   COMMAND QUALIFIERS

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

; Size:         1391 code + 668 data bytes
; Run Time:       00:30.9
; Elapsed Time:   01:19.5
; Lines/CPU Min:   2279
; Lexemes/CPU-Min: 15379
; Memory Used:  154 pages
; Compilation Complete
```

NML_SHOW
LIS

NML_UPGRAD
LIS

NML_V2COMP
LIS

NML_UTIL
LIS