

```

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  LLL
NNNNNN  NNN  MMM      MMM  LLL
NNNNNN  NNN  MMM      MMM  LLL
NNN      NNN  NNN  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      NNN  MMM      MMM  LLL
NNN      NNN  MMM      MMM  LLL
NNN      NNN  MMM      MMM  LLL
NNN      NNN  MMM      MMM  LLLLLLLLLLLLLLLLLL
NNN      NNN  MMM      MMM  LLLLLLLLLLLLLLLLLL
NNN      NNN  MMM      MMM  LLLLLLLLLLLLLLLLLL

```

_S

Ps

--

NP

NP

SG

SOI

NP

PA

-L

```

NN      NN  MM      MM  LL      RRRRRRRR  EEEEEEEEE  AAAAAA  DDDDDDD  DD
NN      NN  MM      MM  LL      RRRRRRRR  EEEEEEEEE  AAAAAA  DDDDDDD  DD
NN      NN  MMMM    MMMM LL      RR      RR  EE      AA      AA  DD      DD
NN      NN  MMMM    MMMM LL      RR      RR  EE      AA      AA  DD      DD
NNNN    NN  MM      MM  LL      RR      RR  EE      AA      AA  DD      DD
NNNN    NN  MM      MM  LL      RR      RR  EE      AA      AA  DD      DD
NN  NN  NN  MM      MM  LL      RRRRRRRR  EEEEEEEEE  AA      AA  DD      DD
NN  NN  NN  MM      MM  LL      RRRRRRRR  EEEEEEEEE  AA      AA  DD      DD
NN      NNNN  MM      MM  LL      RR  RR  EE      AAAAAAAAAA  DD      DD
NN      NNNN  MM      MM  LL      RR  RR  EE      AAAAAAAAAA  DD      DD
NN      NN  MM      MM  LL      RR      RR  EE      AA      AA  DD      DD
NN      NN  MM      MM  LL      RR      RR  EE      AA      AA  DD      DD
NN      NN  MM      MM  LLLLLLLLLL  RR      RR  EEEEEEEEE  AA      AA  DDDDDDD  DD
NN      NN  MM      MM  LLLLLLLLLL  RR      RR  EEEEEEEEE  AA      AA  DDDDDDD  DD

```

```

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 Read information module'
2 0002 0 MODULE NML$READ (
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 Network Management Listener
36 0036 1
37 0037 1 ABSTRACT:
38 0038 1
39 0039 1 These routines dispatch NICE READ function requests to the entity
40 0040 1 processing routine.
41 0041 1
42 0042 1 ENVIRONMENT: VAX/VMS Operating System
43 0043 1
44 0044 1 AUTHOR: Kathy Perko
45 0045 1
46 0046 1 CREATION DATE: 30-June-1982
47 0047 1
48 0048 1 MODIFIED BY:
49 0049 1
50 0050 1 V03-007 MKP0007 Kathy Perko 7-Jan-1984
51 0051 1 Add X25 Access Module entity.
52 0052 1
53 0053 1 V03-006 MKP0006 Kathy Perko 4-Aug-1983
54 0054 1 Enhance node permanent database to use more ISAM keys - to
55 0055 1 speed it up.
56 0056 1
57 0057 1 V03-005 MKP0005 Kathy Perko 28-April-1983

```

:	58	0058	1	:		Add Adjacent service circuit database.
:	59	0059	1	:		
:	60	0060	1	:	V03-004	MKP0004 Kathy Perko 17-Jan-1983
:	61	0061	1	:		Add support for the NI Configurator Module.
:	62	0062	1	:		
:	63	0063	1	:	V03-003	MKP0003 Kathy Perko 14-Nov-1982
:	64	0064	1	:		Add circuit qualifier to SHOW KNOWN and ACTIVE NODES.
:	65	0065	1	:		
:	66	0066	1	:	V03-002	MKP0002 Kathy Perko 8-Nov-1982
:	67	0067	1	:		Add area entity.
:	68	0068	1	:		
:	69	0069	1	:	V03-001	MKP0001 Kathy Perko 5-Oct-1982
:	70	0070	1	:		Add show for adjacent nodes, and for circuits adjacent nodes.
:	71	0071	1	:	--	
:	72	0072	1	:		

```
74 0073 1 %SBTTL 'Declarations'
75 0074 1
76 0075 1
77 0076 1 !! TABLE OF CONTENTS:
78 0077 1 !!
79 0078 1
80 0079 1 FORWARD ROUTINE
81 0080 1     NML$READ           : NOVALUE,
82 0081 1     NML_READ       : NOVALUE,
83 0082 1     NML_READ_LOGGING : NOVALUE,
84 0083 1     NML_READ_NODE   : NOVALUE,
85 0084 1     NML_READ_EXECUTOR : NOVALUE,
86 0085 1     NML_READ_NETWORK : NOVALUE,
87 0086 1     NML_READ_AREA   : NOVALUE,
88 0087 1     NML_SHOW_LINKS  : NOVALUE,
89 0088 1     NML_READ_PLURAL  : NOVALUE;
90 0089 1
91 0090 1 !!
92 0091 1 !! INCLUDE FILES:
93 0092 1 !!
94 0093 1
95 0094 1 LIBRARY 'LIBS:NMLLIB.L32';
96 0095 1 LIBRARY 'SHRLIBS:NMALIBRY.L32';
97 0096 1 LIBRARY 'SYS$LIBRARY:STARLET.L32';
98 0097 1
99 0098 1 !!
100 0099 1 !! EXTERNAL REFERENCES:
101 0100 1 !!
102 0101 1
103 0102 1 $NML_EXTDEF;
104 0103 1
105 0104 1 EXTERNAL ROUTINE
106 0105 1     LIB$ESTABLISH : ADDRESSING_MODE (GENERAL),
107 0106 1     LIB$REVERT   : ADDRESSING_MODE (GENERAL),
108 0107 1     NML$OPENFILE,
109 0108 1     NML$H_D_REPLY,
110 0109 1     NML$MAIRHANDLER,
111 0110 1     NML$ERROR_1,
112 0111 1     NML$ERROR_2,
113 0112 1     NML$LISTENTITY,
114 0113 1     NML$LIST_KNOWN_NODES,
115 0114 1     NML$LIST_TYPE_NODES,
116 0115 1     NML$LISTKNOWN,
117 0116 1     NML$READACTLOG,
118 0117 1     NML$READKNOLOG,
119 0118 1     NML$READLOGGING,
120 0119 1     NML$SEND,
121 0120 1     NML$SHOWMULTIPLE,
122 0121 1     NML$SHOW_MULTIPLE_NODES,
123 0122 1     NML$SHOW_KNOWN_LOOP,
124 0123 1     NML$SHOWENTITY,
125 0124 1     NML$SHOW_CIRCUIT,
126 0125 1     NML$SHOWNODEBYNAME,
127 0126 1     NML$SHOWEXECUTOR,
128 0127 1     NML$CALL_NI_CONFIG;
```

```

130 0128 1 |
131 0129 1 | Macro to build dispatch table for an entity.
132 0130 1 |
133 0131 1 | MACRO $TAB (TAB,
134 0132 1 |     DISPATCH_RTN,
135 0133 1 |     SHOW_RTN,
136 0134 1 |     SHOW_W_QUAL_RTN,
137 0135 1 |     SHOW_KNO_RTN,
138 0136 1 |     SHOW_KNO_W_QUAL_RTN,
139 0137 1 |     SHOW_ACT_RTN,
140 0138 1 |     LIST_RTN,
141 0139 1 |     LIST_W_QUAL_RTN,
142 0140 1 |     LIST_KNO_RTN,
143 0141 1 |     LIST_KNO_W_QUAL_RTN) =
144 0142 1 |
145 0143 1 |     OWN TAB : BBLOCK [%LENGTH * 4] INITIAL (
146 0144 1 |     $PIC (DISPATCH_RTN, TAB),
147 0145 1 |     $PIC (SHOW_RTN, TAB),
148 0146 1 |     $PIC (SHOW_W_QUAL_RTN, TAB),
149 0147 1 |     $PIC (SHOW_KNO_RTN, TAB),
150 0148 1 |     $PIC (SHOW_KNO_W_QUAL_RTN, TAB),
151 0149 1 |     $PIC (SHOW_ACT_RTN, TAB),
152 0150 1 |     $PIC (LIST_RTN, TAB),
153 0151 1 |     $PIC (LIST_W_QUAL_RTN, TAB),
154 0152 1 |     $PIC (LIST_KNO_RTN, TAB),
155 0153 1 |     $PIC (LIST_KNO_W_QUAL_RTN, TAB),
156 0154 1 |     0) !There is never a LIST ACTIVE command.
157 0155 1 |     %,
158 0156 1 |
159 0157 1 |     $PIC (ADDR, TAB) =
160 0158 1 |     %IF %IDENTICAL (ADDR, 0)
161 0159 1 |     %THEN LONG (0)
162 0160 1 |     %ELSE LONG (%NAME (ADDR) - %NAME (TAB))
163 0161 1 |     %FI
164 0162 1 |     %;
165 0163 1 |
166 0164 1 |
167 0165 1 |
168 0166 1 | Dispatch tables. There is one table for each internal NML entity (NML
169 0167 1 | internal entities are broken down more than NICE entities). The table
170 0168 1 | specifies the following information about the entity:
171 0169 1 |     The address of the dispatch routine in this module for the entity.
172 0170 1 |     The dispatch routines vary depending on the different
173 0171 1 |     formats the entities can have.
174 0172 1 |     The addresses of the routines which perform the requested read:
175 0173 1 |     - Show single entity
176 0174 1 |     - Show single entity with specified qualifier
177 0175 1 |     - Show known entities
178 0176 1 |     - Show known entities with specified qualifier
179 0177 1 |     - List single entity
180 0178 1 |     - List single entity with specified qualifier
181 0179 1 |     - List known entities
182 0180 1 |     - List known entities with specified qualifier
183 0181 1 |     - Show active entities
184 0182 1 |
185 0183 1 |
186 P 0184 1 | $TAB (LINE_TAB,                                ! NML$C_LINE

```

```

187 P 0185 1 NML_READ,
188 PP 0186 1 NML$SHOWENTITY, 0, NML$SHOWMULTIPLE, 0,
189 P 0187 1 NML$SHOWMULTIPLE,
190 0188 1 NML$LISTENTITY, 0, NML$LISTKNOWN, 0);
191 0189 1
192 P 0190 1 $TAB (LOGGING_TAB, ! NML$C_LOGGING
193 P 0191 1 NML_READ_LOGGING,
194 PP 0192 1 NML$READLOGGING, 0, NML$READKNOLOG, 0,
195 P 0193 1 NML$READACTLOG,
196 0194 1 NML$READLOGGING, 0, NML$READKNOLOG, 0);
197 0195 1
198 0196 1 BIND SINK_TAB = UPLIT (0);
199 0197 1
200 P 0198 1 $TAB (NODE_TAB, ! NML$C_NODE
201 PP 0199 1 NML_READ_NODE,
202 PP 0200 1 NML$SHOWENTITY, 0, NML$SHOW_MULTIPLE_NODES,
203 PP 0201 1 NML$SHOW_MULTIPLE_NODES,
204 P 0202 1 NML$SHOW_MULTIPLE_NODES,
205 0203 1 NML$LISTENTITY, 0, NML$LIST_KNOWN_NODES, 0);
206 0204 1
207 P 0205 1 $TAB (NODEBYNAME_TAB, ! NML$C_NODEBYNAME
208 P 0206 1 NML_READ_NODE,
209 PP 0207 1 NML$SHOWNODEBYNAME, 0, NML$SHOW_MULTIPLE_NODES, 0,
210 P 0208 1 NML$SHOW_MULTIPLE_NODES,
211 0209 1 NML$LISTENTITY, 0, NML$LIST_KNOWN_NODES, 0);
212 0210 1
213 P 0211 1 $TAB (LOOPNODE_TAB, ! NML$C_LOOPNODE
214 PP 0212 1 NML_READ_NODE,
215 PP 0213 1 0, 0, NML$SHOW_KNOWN_LOOP, 0,
216 P 0214 1 0,
217 0215 1 0, 0, NML$LIST_TYPE_NODES, 0);
218 0216 1
219 P 0217 1 $TAB (ADJACENT_TAB, ! NML$C_ADJACENT_NODE
220 PP 0218 1 NML_READ_NODE,
221 P 0219 1 0, 0, NML$SHOWMULTIPLE, NML$SHOWMULTIPLE,
222 P 0220 1 0,
223 0221 1 0, 0, 0);
224 0222 1
225 P 0223 1 $TAB (EXECUTOR_TAB, ! NML$C_EXECUTOR
226 PP 0224 1 NML_READ_EXECUTOR,
227 P 0225 1 NML$SHOWEXECUTOR, 0, 0, 0,
228 P 0226 1 0,
229 0227 1 NML$LISTENTITY, 0, 0, 0);
230 0228 1
231 P 0229 1 $TAB (OBJECT_TAB, ! NML$C_OBJECT
232 PP 0230 1 NML_READ,
233 P 0231 1 NML$SHOWENTITY, 0, NML$SHOWMULTIPLE, 0,
234 P 0232 1 NML$SHOWMULTIPLE,
235 0233 1 NML$LISTENTITY, 0, NML$LISTKNOWN, 0);
236 0234 1
237 P 0235 1 $TAB (CIRCUIT_TAB, ! NML$C_CIRCUIT
238 PP 0236 1 NML_READ,
239 P 0237 1 NML$SHOW_CIRCUIT, NML$SHOW_CIRCUIT, NML$SHOWMULTIPLE, NML$SHOWMULTIPLE,
240 P 0238 1 NML$SHOWMULTIPLE,
241 0239 1 NML$LISTENTITY, 0, NML$LISTKNOWN, 0);
242 0240 1
243 0241 1 BIND CIRCUIT_ADJACENT_TAB = UPLIT (0);

```

```

244 0242 1
245 0243 1 BIND CIRCUIT_ADJ_SRV_TAB = UPLIT (0);
246 0244 1
247 P 0245 1 $TAB (AREA_TAB, ! NML$C_AREA
248 P 0246 1 NML_READ_AREA,
249 P 0247 1 NML$SHOWENTITY, 0, NML$SHOWMULTIPLE, 0,
250 P 0248 1 NML$SHOWMULTIPLE,
251 0249 1 0, 0, 0);
252 0250 1
253 0251 1
254 P 0252 1 $TAB (ACCESS_TAB, ! NML$C_X25_ACCESS
255 P 0253 1 NML_READ_NETWORK,
256 P 0254 1 NML$SHOWENTITY, 0, NML$SHOWMULTIPLE, 0,
257 P 0255 1 0,
258 0256 1 NML$LISTENTITY, 0, NML$LISTKNOWN, 0);
259 0257 1
260 P 0258 1 $TAB (PROT_NET_TAB, ! NML$C_PROT_NET_TAB
261 P 0259 1 NML_READ_NETWORK,
262 P 0260 1 NML$SHOWENTITY, 0, 0, 0,
263 P 0261 1 0,
264 0262 1 NML$LISTENTITY, 0, 0, 0);
265 0263 1
266 P 0264 1 $TAB (PROT_DTE_TAB, ! NML$C_PROT_DTE
267 P 0265 1 NML_READ,
268 P 0266 1 NML$SHOWENTITY, 0, NML$SHOWMULTIPLE, 0,
269 P 0267 1 NML$SHOWMULTIPLE,
270 0268 1 NML$LISTENTITY, 0, NML$LISTKNOWN, 0);
271 0269 1
272 P 0270 1 $TAB (PROT_GRP_TAB, ! NML$C_PROT_GRP
273 P 0271 1 NML_READ,
274 P 0272 1 NML$SHOWMULTIPLE, NML$SHOWMULTIPLE, NML$SHOWMULTIPLE, NML$SHOWMULTIPLE,
275 P 0273 1 0,
276 0274 1 NML$LISTKNOWN, NML$LISTKNOWN, NML$LISTKNOWN, NML$LISTKNOWN);
277 0275 1
278 P 0276 1 $TAB (SERV_TAB, ! NML$C_X25_SERV and NML$C_X29_SERV
279 P 0277 1 NML_READ,
280 P 0278 1 NML$SHOWENTITY, 0, 0, 0,
281 P 0279 1 0,
282 0280 1 NML$LISTENTITY, 0, 0, 0);
283 0281 1
284 P 0282 1 $TAB (SERV_DEST_TAB, ! NML$C_X25_SERV_DEST and NML$C_X29_SERV_DEST
285 P 0283 1 NML_READ,
286 P 0284 1 NML$SHOWENTITY, 0, NML$SHOWMULTIPLE, 0,
287 P 0285 1 NML$SHOWMULTIPLE,
288 0286 1 NML$LISTENTITY, 0, NML$LISTKNOWN, 0);
289 0287 1
290 P 0288 1 $TAB (TRACE_TAB, ! NML$C_TRACE
291 P 0289 1 NML_READ,
292 P 0290 1 NML$SHOWENTITY, 0, 0, 0,
293 P 0291 1 0,
294 0292 1 NML$LISTENTITY, 0, 0, 0);
295 0293 1
296 P 0294 1 $TAB (TRACEPNT_TAB, ! NML$C_TRACEPNT
297 P 0295 1 NML_READ,
298 P 0296 1 NML$SHOWENTITY, 0, NML$SHOWMULTIPLE, 0,
299 P 0297 1 NML$SHOWMULTIPLE,
300 0298 1 NML$LISTENTITY, 0, NML$LISTKNOWN, 0);

```



```

301 0299 1
302 P 0300 1 $TAB (NI CONFIG TAB,          ! NML$C_NI_CONFIG
303 P 0301 1      NML_READ,
304 P 0302 1      NML$CALL_NI_CONFIG,    0,      NML$CALL_NI_CONFIG,    0,
305 P 0303 1      0,
306 P 0304 1      NML$LISTENTITY,        0,      NML$LISTKNOWN,        0);
307 0305 1
308 P 0306 1 $TAB (LINK TAB,              ! NML$C_LINK
309 P 0307 1      NML_SHOW_LINKS,
310 P 0308 1      NML$SHOWENTITY,        0,      NML$SHOWMULTIPLE,    NML$SHOWMULTIPLE,
311 P 0309 1      0,
312 0310 1      0,      0,      0);
313 0311 1
314 0312 1
315 0313 1
316 0314 1 ! Table table.  Contains pointers to Dispatch tables for NML entities.
317 0315 1 ! Indexed by NML$C_entity definitions.
318 0316 1
319 0317 1 OWN TABLE TAB : VECTOR [NML$C_MAXENTITY] INITIAL (
320 0318 1     $PIC (LINE_TAB, TABLE_TAB),
321 0319 1     $PIC (LOGGING_TAB, TABLE_TAB),
322 0320 1     $PIC (SINK_TAB, TABLE_TAB),
323 0321 1     $PIC (NODE_TAB, TABLE_TAB),
324 0322 1     $PIC (NODEBYNAME_TAB, TABLE_TAB),
325 0323 1     $PIC (LOOPNODE_TAB, TABLE_TAB),
326 0324 1     $PIC (ADJACENT_TAB, TABLE_TAB),
327 0325 1     $PIC (EXECUTOR_TAB, TABLE_TAB),
328 0326 1     $PIC (OBJECT_TAB, TABLE_TAB),
329 0327 1     $PIC (CIRCUIT_TAB, TABLE_TAB),
330 0328 1     $PIC (CIRCUIT-ADJACENT_TAB, TABLE_TAB),
331 0329 1     $PIC (CIRCUIT-ADJ SRV TAB, TABLE_TAB),
332 0330 1     $PIC (AREA_TAB, TABLE_TAB),
333 0331 1     $PIC (ACCESS_TAB, TABLE_TAB),          ! X25
334 0332 1     $PIC (PROT_NET_TAB, TABLE_TAB),
335 0333 1     $PIC (PROT-DTE_TAB, TABLE_TAB),
336 0334 1     $PIC (PROT-GRP_TAB, TABLE_TAB),
337 0335 1     $PIC (SERV_TAB, TABLE TAB),           ! X25
338 0336 1     $PIC (SERV-DEST_TAB, TABLE TAB),     ! X25
339 0337 1     $PIC (TRACE_TAB, TABLE TAB),
340 0338 1     $PIC (TRACEPNT_TAB, TABLE TAB),
341 0339 1     $PIC (SERV_TAB, TABLE TAB),          ! X29
342 0340 1     $PIC (SERV-DEST_TAB, TABLE TAB),     ! X29
343 0341 1     $PIC (NI CONFIG_TAB, TABLE_TAB),
344 0342 1     $PIC (LINK_TAB, TABLE_TAB);

```

```

346 0343 1 %SBTTL 'NML$READ Read parameters main routine'
347 0344 1 GLOBAL ROUTINE NML$READ : NOVALUE =
348 0345 1
349 0346 1 |**
350 0347 1 | FUNCTIONAL DESCRIPTION:
351 0348 1 |
352 0349 1 |     This routine dispatches the NICE read parameters command to
353 0350 1 |     the correct function handler.
354 0351 1 |
355 0352 1 | IMPLICIT INPUTS:
356 0353 1 |
357 0354 1 |     NML$GB_OPTIONS contains the option byte parsed from the NICE message.
358 0355 1 |     NML$GB_ENTITY_CODE contains the entity code.
359 0356 1 |
360 0357 1 | --
361 0358 1
362 0359 2 BEGIN
363 0360 2
364 0361 2 MAP
365 0362 2     NML$GB_ENTITY_FORMAT : BYTE SIGNED,
366 0363 2     NML$GB_OPTIONS : BBLOCK [1];
367 0364 2
368 0365 2 LOCAL
369 0366 2     INDEX,                | Index into Information Table (in NMLDAT)
370 0367 2     ENT_TAB : REF BBLOCK,  | Dispatch table reference
371 0368 2     READ_TABLE_ADR: REF BBLOCK, | Address of SHOW or LIST portion of entity
372 0369 2                               | dispatch table.
373 0370 2     RTN_ADDR,            | Temporary routine address
374 0371 2     READ_RTN:           | Address of routine to perform
375 0372 2                               | read requested by NICE
376 0373 2                               | message.
377 0374 2
378 0375 2
379 0376 2 |
380 0377 2 | Get the index for information type requested by the NICE command.
381 0378 2 |
382 0379 2 SELECTONEU .NML$GB_INFO OF
383 0380 2 SET
384 0381 2 [NMASC_OPINF_SUM]:
385 0382 2     INDEX = NML$C_SUMMARY; ! Summary
386 0383 2 [NMASC_OPINF_STA]:
387 0384 2     INDEX = NML$C_STATUS; ! Status
388 0385 2 [NMASC_OPINF_CHA]:
389 0386 2     INDEX = NML$C_CHARACTERISTICS; ! Characteristics
390 0387 2 [NMASC_OPINF_COU]:
391 0388 2     |
392 0389 2     | Counters are kept only in the volatile data base.
393 0390 2     |
394 0391 2     IF .NML$GB_OPTIONS [NMASV_OPT_PER] THEN
395 0392 2         INDEX = -1
396 0393 2     ELSE
397 0394 2         INDEX = NML$C_COUNTERS;
398 0395 2 [NMASC_OPINF_EVE]:
399 0396 2     INDEX = NML$C_EVENTS; ! Events
400 0397 2 [OTHERWISE]:
401 0398 2     INDEX = -1; ! Option error
402 0399 2 TES;

```

```

403 0400 2 |
404 0401 2 | If the information type is illegal, signal an error response to NCP
405 0402 2 |
406 0403 2 | IF .INDEX EQLU -1 THEN
407 0404 2 |     NML$ERROR_1 (NMA$C_STS_FUN);
408 0405 2 |
409 0406 2 |
410 0407 2 | Get address of entity's dispatch table. The addresses are stored as offsets
411 0408 2 | to make NML$HR PIC. Change the offset into a useable address.
412 0409 2 |
413 0410 2 | ENT_TAB = .TABLE_TAB [.NML$GL_NML_ENTITY] + TABLE_TAB;
414 0411 2 | IF .ENT_TAB NEQA 0 THEN
415 0412 2 |     BEGIN
416 0413 2 |     RTN_ADDR = .ENT_TAB [DT$S_DISPATCH] + .ENT_TAB;
417 0414 2 |
418 0415 2 |     Go to dispatch table for the entity specified in the NICE message.
419 0416 2 |     Get the address of the routine which performs the type of read
420 0417 2 |     requested.
421 0418 2 |
422 0419 3 | IF .RTN_ADDR NEQA .ENT_TAB THEN
423 0420 4 |     BEGIN
424 0421 4 |
425 0422 4 |     Get address of portion of entity's dispatch table containing
426 0423 4 |     the change routine addresses for the function (SHOW or LIST)
427 0424 4 |     specified by the NICE message.
428 0425 4 |
429 0426 4 |     IF .NML$GB OPTIONS [NMA$V_OPT_PER] THEN
430 0427 4 |         READ_TABLE_ADR = ENT_TAB [DT$A_LIST_ROUTINES]
431 0428 4 |     ELSE
432 0429 4 |         READ_TABLE_ADR = ENT_TAB [DT$A_SHOW_ROUTINES];
433 0430 4 |
434 0431 4 |
435 0432 4 |     Each function's portion of the entity's dispatch table contains
436 0433 4 |     the addresses of five read routines. These routines do the
437 0434 4 |     following:
438 0435 4 |     - Read a single entity
439 0436 4 |     - Read a single entity with the specified qualifier
440 0437 4 |     - Read known entities
441 0438 4 |     - Read known entities with the specified qualifier
442 0439 4 |     - Read active entities (never used for LIST because
443 0440 4 |       the permanent data base does not maintain real state info.
444 0441 4 |
445 0442 4 | SELECTONEU .NML$GB_ENTITY_FORMAT OF
446 0443 4 | SET
447 0444 4 |     [NMA$C_ENT_KNO,           | read known
448 0445 4 |     NMA$C_ENT_LOO,           | read loopnodes
449 0446 4 |     NMA$C_ENT_ADJ]:         | read adjacent nodes
450 0447 4 |     IF .NML$GL_PRS_FLGS [NML$V_PRS_QUALIFIER] THEN
451 0448 4 |         READ_RTN = .READ_TABLE_ADR [RD$S_KNOWN_W_QUAL]
452 0449 4 |     ELSE
453 0450 4 |         READ_RTN = .READ_TABLE_ADR [RD$S_KNOWN];
454 0451 4 |     [NMA$C_ENT_ACT]:         | read active
455 0452 4 |     READ_RTN = .READ_TABLE_ADR [RD$S_ACTIVE];
456 0453 4 |     [0]:                     | read active X25-Protocol network,
457 0454 4 |                             | node by address or single area.
458 0455 4 |     READ_RTN = .READ_TABLE_ADR [RD$S_ENTITY];
459 0456 4 |     [1 TO 16]:             | read single entity

```

```

460 0457 4          IF .NMLSGL_PRS_FLGS [NMLS$V_PRS_QUALIFIER] THEN
461 0458 4          READ_RTN = .READ_TABLE_ADR [RDSL_ENTITY_W_QUAL]
462 0459 4          ELSE
463 0460 4          READ_RTN = .READ_TABLE_ADR [RDSL_ENTITY];
464 0461 4          [OTHERWISE]:
465 0462 4          READ_RTN = 0;          ! Invalid entity format for SHOW
466 0463 4          TES;
467 0464 4          !
468 0465 4          ! The routine addresses are stored as offsets (to make NMLSHR PIC).
469 0466 4          ! Make the offset into a callable routine address.
470 0467 4          !
471 0468 4          IF .READ_RTN NEQ 0 THEN
472 0469 5          BEGIN
473 0470 5          READ_RTN = .READ_RTN + .ENT_TAB;
474 0471 5          !
475 0472 5          ! Call read routine.
476 0473 5          !
477 0474 5          (.RTN_ADDR) (.NMLSGL_NML_ENTITY,
478 0475 5          .INDEX,
479 0476 5          .READ_RTN);
480 0477 5          END
481 0478 4          ELSE
482 0479 4          NML$ERROR_2 (NMASC_STS_IDE, .NMLSGB_ENTITY_FORMAT);
483 0480 4          END
484 0481 3          ELSE
485 0482 3          NML$ERROR_2 (NMASC_STS_IDE, .NMLSGB_ENTITY_FORMAT);
486 0483 3          END
487 0484 2          ELSE
488 0485 2          NML$ERROR_2 (NMASC_STS_IDE, .NMLSGB_ENTITY_FORMAT);
489 0486 1          END;          ! End of NMLSREAD

```

.TITLE NMLSREAD NML Read information module
.IDENT \V04-000\

.PSECT \$PLITS,NOWRT,NOEXE,2

00000000 00000 P.AAA: .LONG 0
00000000 00004 P.AAB: .LONG 0
00000000 00008 P.AAC: .LONG 0

.PSECT \$OWNS,NOEXE,2

00000000V 00000 LINE_TAB:
00000000* 00004 .LONG <NML_READ-LINE_TAB>
00000000 00008 .LONG <NML\$SHOWENTITY-LINE_TAB>
00000000* 0000C .LONG 0
00000000 00010 .LONG <NML\$SHOWMULTIPLE-LINE_TAB>
00000000* 00014 .LONG 0
00000000* 00018 .LONG <NML\$SHOWMULTIPLE-LINE_TAB>
00000000* 0001C .LONG <NML\$LISTENTITY-LINE_TAB>
00000000 00020 .LONG 0
00000000* 00024 .LONG <NML\$LISTKNOWN-LINE_TAB>
00000000 00028 .LONG 0
00000000V 0002C LOGGING_TAB:
.LONG <NML_READ_LOGGING-LOGGING_TAB>

```
00000000* 00030 .LONG <NMLSREADLOGGING-LOGGING_TAB>
00000000 00034 .LONG 0
00000000* 00038 .LONG <NMLSREADKNOLOG-LOGGING_TAB>
00000000 0003C .LONG 0
00000000* 00040 .LONG <NMLSREADACTLOG-LOGGING_TAB>
00000000* 00044 .LONG <NMLSREADLOGGING-LOGGING_TAB>
00000000 00048 .LONG 0
00000000* 0004C .LONG <NMLSREADKNOLOG-LOGGING_TAB>
00000000 00050 .LONG 0
00000000 00054 .LONG 0
00000000V 00058 NODE_TAB:
                                .LONG <NML_READ_NODE-NODE_TAB>
                                .LONG <NML$SHOWENTITY-NODE_TAB>
00000000* 0005C .LONG 0
00000000 00060 .LONG <NML$SHOW_MULTIPLE_NODES-NODE_TAB>
00000000* 00064 .LONG <NML$SHOW_MULTIPLE_NODES-NODE_TAB>
00000000* 00068 .LONG <NML$SHOW_MULTIPLE_NODES-NODE_TAB>
00000000* 0006C .LONG <NML$SHOW_MULTIPLE_NODES-NODE_TAB>
00000000* 00070 .LONG <NML$LISTENTITY-NODE_TAB>
00000000 00074 .LONG 0
00000000* 00078 .LONG <NML$LIST_KNOWN_NODES-NODE_TAB>
00000000 0007C .LONG 0
00000000 00080 .LONG 0
00000000V 00084 NODEBYNAME_TAB:
                                .LONG <NML_READ_NODE-NODEBYNAME_TAB>
                                .LONG <NML$SHOWNODEBYNAME-NODEBYNAME_TAB>
00000000* 00088 .LONG 0
00000000 0008C .LONG <NML$SHOW_MULTIPLE_NODES-NODEBYNAME_TAB>
00000000* 00090 .LONG 0
00000000 00094 .LONG <NML$SHOW_MULTIPLE_NODES-NODEBYNAME_TAB>
00000000* 00098 .LONG <NML$SHOW_MULTIPLE_NODES-NODEBYNAME_TAB>
00000000* 0009C .LONG <NML$LISTENTITY-NODEBYNAME_TAB>
00000000 000A0 .LONG 0
00000000* 000A4 .LONG <NML$LIST_KNOWN_NODES-NODEBYNAME_TAB>
00000000 000A8 .LONG 0
00000000 000AC .LONG 0
00000000V 000B0 LOOPNODE_TAB:
                                .LONG <NML_READ_NODE-LOOPNODE_TAB>
00000000 000B4 .LONG 0
00000000 000B8 .LONG 0
00000000* 000BC .LONG <NML$SHOW_KNOWN_LOOP-LOOPNODE_TAB>
00000000 000C0 .LONG 0
00000000 000C4 .LONG 0
00000000 000C8 .LONG 0
00000000 000CC .LONG 0
00000000* 000D0 .LONG <NML$LIST_TYPE_NODES-LOOPNODE_TAB>
00000000 000D4 .LONG 0
00000000 000D8 .LONG 0
00000000V 000DC ADJACENT_TAB:
                                .LONG <NML_READ_NODE-ADJACENT_TAB>
00000000 000E0 .LONG 0
00000000 000E4 .LONG 0
00000000* 000E8 .LONG <NML$SHOWMULTIPLE-ADJACENT_TAB>
00000000* 000EC .LONG <NML$SHOWMULTIPLE-ADJACENT_TAB>
00000000 000F0 .LONG 0
00000000 000F4 .LONG 0
00000000 000F8 .LONG 0
00000000 000FC .LONG 0
00000000 00100 .LONG 0
```

```
00000000 00104 .LONG 0
00000000V 00108 EXECUTOR_TAB:
      .LONG <NML READ EXECUTOR-EXECUTOR_TAB>
00000000* 0010C .LONG <NML$SHOWEXECUTOR-EXECUTOR_TAB>
00000000 00110 .LONG 0
00000000 00114 .LONG 0
00000000 00118 .LONG 0
00000000 0011C .LONG 0
00000000* 00120 .LONG <NML$LISTENTITY-EXECUTOR_TAB>
00000000 00124 .LONG 0
00000000 00126 .LONG 0
00000000 0012C .LONG 0
00000000 00130 .LONG 0
00000000V 00134 OBJECT_TAB:
      .LONG <NML READ-OBJECT_TAB>
00000000* 00138 .LONG <NML$SHOWENTITY-OBJECT_TAB>
00000000 0013C .LONG 0
00000000* 00140 .LONG <NML$SHOWMULTIPLE-OBJECT_TAB>
00000000 00144 .LONG 0
00000000* 00148 .LONG <NML$SHOWMULTIPLE-OBJECT_TAB>
00000000* 0014C .LONG <NML$LISTENTITY-OBJECT_TAB>
00000000 00150 .LONG 0
00000000* 00154 .LONG <NML$LISTKNOWN-OBJECT_TAB>
00000000 00158 .LONG 0
00000000 0015C .LONG 0
00000000V 00160 CIRCUIT_TAB:
      .LONG <NML READ-CIRCUIT_TAB>
00000000* 00164 .LONG <NML$SHOW_CIRCUIT-CIRCUIT_TAB>
00000000* 00168 .LONG <NML$SHOW_CIRCUIT-CIRCUIT_TAB>
00000000* 0016C .LONG <NML$SHOWMULTIPLE-CIRCUIT_TAB>
00000000* 00170 .LONG <NML$SHOWMULTIPLE-CIRCUIT_TAB>
00000000* 00174 .LONG <NML$SHOWMULTIPLE-CIRCUIT_TAB>
00000000* 00178 .LONG <NML$LISTENTITY-CIRCUIT_TAB>
00000000 0017C .LONG 0
00000000* 00180 .LONG <NML$LISTKNOWN-CIRCUIT_TAB>
00000000 00184 .LONG 0
00000000 00188 .LONG 0
00000000V 0018C AREA_TAB:
      .LONG <NML READ AREA-AREA_TAB>
00000000* 00190 .LONG <NML$SHOWENTITY-AREA_TAB>
00000000 00194 .LONG 0
00000000* 00198 .LONG <NML$SHOWMULTIPLE-AREA_TAB>
00000000 0019C .LONG 0
00000000* 001A0 .LONG <NML$SHOWMULTIPLE-AREA_TAB>
00000000 001A4 .LONG 0
00000000 001A8 .LONG 0
00000000 001AC .LONG 0
00000000 001B0 .LONG 0
00000000 001B4 .LONG 0
00000000V 001B8 ACCESS_TAB:
      .LONG <NML READ NETWORK-ACCESS_TAB>
00000000* 001BC .LONG <NML$SHOWENTITY-ACCESS_TAB>
00000000 001C0 .LONG 0
00000000* 001C4 .LONG <NML$SHOWMULTIPLE-ACCESS_TAB>
00000000 001C8 .LONG 0
00000000 001CC .LONG 0
00000000* 001D0 .LONG <NML$LISTENTITY-ACCESS_TAB>
```

```
00000000 001D4 .LONG 0
00000000* 001D8 .LONG <NML$LISTKNOWN-ACCESS_TAB>
00000000 001DC .LONG 0
00000000 001E0 .LONG 0
00000000V 001E4 PROT_NET_TAB:
      .LONG <NML_READ_NETWORK-PROT_NET_TAB>
00000000* 001E8 .LONG <NML$SHOWENTITY-PROT_NET_TAB>
00000000 001EC .LONG 0
00000000 001F0 .LONG 0
00000000 001F4 .LONG 0
00000000 001F8 .LONG 0
00000000* 001FC .LONG <NML$LISTENTITY-PROT_NET_TAB>
00000000 00200 .LONG 0
00000000 00204 .LONG 0
00000000 00208 .LONG 0
00000000 0020C .LONG 0
00000000V 00210 PROT_DTE_TAB:
      .LONG <NML_READ-PROT_DTE_TAB>
00000000* 00214 .LONG <NML$SHOWENTITY-PROT_DTE_TAB>
00000000 00218 .LONG 0
00000000* 0021C .LONG <NML$SHOWMULTIPLE-PROT_DTE_TAB>
00000000 00220 .LONG 0
00000000* 00224 .LONG <NML$SHOWMULTIPLE-PROT_DTE_TAB>
00000000* 00228 .LONG <NML$LISTENTITY-PROT_DTE_TAB>
00000000 0022C .LONG 0
00000000* 00230 .LONG <NML$LISTKNOWN-PROT_DTE_TAB>
00000000 00234 .LONG 0
00000000 00238 .LONG 0
00000000V 0023C PROT_GRP_TAB:
      .LONG <NML_READ-PROT_GRP_TAB>
00000000* 00240 .LONG <NML$SHOWMULTIPLE-PROT_GRP_TAB>
00000000* 00244 .LONG <NML$SHOWMULTIPLE-PROT_GRP_TAB>
00000000* 00248 .LONG <NML$SHOWMULTIPLE-PROT_GRP_TAB>
00000000* 0024C .LONG <NML$SHOWMULTIPLE-PROT_GRP_TAB>
00000000 00250 .LONG 0
00000000* 00254 .LONG <NML$LISTKNOWN-PROT_GRP_TAB>
00000000* 00258 .LONG <NML$LISTKNOWN-PROT_GRP_TAB>
00000000* 0025C .LONG <NML$LISTKNOWN-PROT_GRP_TAB>
00000000* 00260 .LONG <NML$LISTKNOWN-PROT_GRP_TAB>
00000000 00264 .LONG 0
00000000V 00268 SERV_TAB:
      .LONG <NML_READ-SERV_TAB>
00000000* 0026C .LONG <NML$SHOWENTITY-SERV_TAB>
00000000 00270 .LONG 0
00000000 00274 .LONG 0
00000000 00278 .LONG 0
00000000 0027C .LONG 0
00000000* 00280 .LONG <NML$LISTENTITY-SERV_TAB>
00000000 00284 .LONG 0
00000000 00288 .LONG 0
00000000 0028C .LONG 0
00000000 00290 .LONG 0
00000000V 00294 SERV_DEST_TAB:
      .LONG <NML_READ-SERV_DEST_TAB>
00000000* 00298 .LONG <NML$SHOWENTITY-SERV_DEST_TAB>
00000000 0029C .LONG 0
00000000* 002A0 .LONG <NML$SHOWMULTIPLE-SERV_DEST_TAB>
```

```
00000000 002A4 .LONG 0
00000000* 002A8 .LONG <NML$SHOWMULTIPLE-SERV_DEST_TAB>
00000000* 002AC .LONG <NML$LISTENTITY-SERV_DEST_TAB>
00000000 002B0 .LONG 0
00000000* 002B4 .LONG <NML$LISTKNOWN-SERV_DEST_TAB>
00000000 002B8 .LONG 0
00000000 002BC .LONG 0
00000000V 002C0 TRACE_TAB:
      .LONG <NML_READ-TRACE_TAB>
00000000* 002C4 .LONG <NML$SHOWENTITY-TRACE_TAB>
00000000 002C8 .LONG 0
00000000 002CC .LONG 0
00000000 002D0 .LONG 0
00000000 002D4 .LONG 0
00000000* 002D8 .LONG <NML$LISTENTITY-TRACE_TAB>
00000000 002DC .LONG 0
00000000 002E0 .LONG 0
00000000 002E4 .LONG 0
00000000 002E8 .LONG 0
00000000V 002EC TRACEPNT_TAB:
      .LONG <NML_READ-TRACEPNT_TAB>
00000000* 002F0 .LONG <NML$SHOWENTITY-TRACEPNT_TAB>
00000000 002F4 .LONG 0
00000000* 002F8 .LONG <NML$SHOWMULTIPLE-TRACEPNT_TAB>
00000000 002FC .LONG 0
00000000* 00300 .LONG <NML$SHOWMULTIPLE-TRACEPNT_TAB>
00000000* 00304 .LONG <NML$LISTENTITY-TRACEPNT_TAB>
00000000 00308 .LONG 0
00000000* 0030C .LONG <NML$LISTKNOWN-TRACEPNT_TAB>
00000000 00310 .LONG 0
00000000 00314 .LONG 0
00000000V 00318 NI_CONFIG_TAB:
      .LONG <NML_READ-NI_CONFIG_TAB>
00000000* 0031C .LONG <NML$CALL_NI_CONFIG-NI_CONFIG_TAB>
00000000 00320 .LONG 0
00000000* 00324 .LONG <NML$CALL_NI_CONFIG-NI_CONFIG_TAB>
00000000 00328 .LONG 0
00000000 0032C .LONG 0
00000000* 00330 .LONG <NML$LISTENTITY-NI_CONFIG_TAB>
00000000 00334 .LONG 0
00000000* 00338 .LONG <NML$LISTKNOWN-NI_CONFIG_TAB>
00000000 0033C .LONG 0
00000000 00340 .LONG 0
00000000V 00344 LINK_TAB:
      .LONG <NML_SHOW_LINKS-LINK_TAB>
00000000* 00348 .LONG <NML$SHOWENTITY-LINK_TAB>
00000000 0034C .LONG 0
00000000* 00350 .LONG <NML$SHOWMULTIPLE-LINK_TAB>
00000000* 00354 .LONG <NML$SHOWMULTIPLE-LINK_TAB>
00000000 00358 .LONG 0
00000000 0035C .LONG 0
00000000 00360 .LONG 0
00000000 00364 .LONG 0
00000000 00368 .LONG 0
00000000 0036C .LONG 0
00000000* 00370 TABLE_TAB:
      .LONG <LINE_TAB-TABLE_TAB>
```



```

00000000* 00374 .LONG <LOGGING_TAB-TABLE_TAB>
00000000* 00378 .LONG <SINK_TAB-TABLE_TAB>
00000000* 0037C .LONG <NODE_TAB-TABLE_TAB>
00000000* 00380 .LONG <NODEBYNAME_TAB-TABLE_TAB>
00000000* 00384 .LONG <LOOPNODE_TAB-TABLE_TAB>
00000000* 00388 .LONG <ADJACENT_TAB-TABLE_TAB>
00000000* 0038C .LONG <EXECUTOR_TAB-TABLE_TAB>
00000000* 00390 .LONG <OBJECT_TAB-TABLE_TAB>
00000000* 00394 .LONG <CIRCUIT_TAB-TABLE_TAB>
00000000* 00398 .LONG <CIRCUIT_ADJACENT_TAB-TABLE_TAB>
00000000* 0039C .LONG <CIRCUIT_ADJ_SRV_TAB-TABLE_TAB>
00000000* 003A0 .LONG <AREA_TAB-TABLE_TAB>
00000000* 003A4 .LONG <ACCESS_TAB-TABLE_TAB>
00000000* 003A8 .LONG <PROT_NET_TAB-TABLE_TAB>
00000000* 003AC .LONG <PROT_DTE_TAB-TABLE_TAB>
00000000* 003B0 .LONG <PROT_GRP_TAB-TABLE_TAB>
00000000* 003B4 .LONG <SERV_TAB-TABLE_TAB>
00000000* 003B8 .LONG <SERV_DEST_TAB-TABLE_TAB>
00000000* 003BC .LONG <TRACE_TAB-TABLE_TAB>
00000000* 003C0 .LONG <TRACEPNT_TAB-TABLE_TAB>
00000000* 003C4 .LONG <SERV_TAB-TABLE_TAB>
00000000* 003C8 .LONG <SERV_DEST_TAB-TABLE_TAB>
00000000* 003CC .LONG <NI_CONFIG_TAB-TABLE_TAB>
00000000* 003D0 .LONG <LINK_TAB-TABLE_TAB>
003D4 .BLKB 4

```

```

SINK_TAB= P.AAA
CIRCOIT_ADJACENT_TAB= P.AAB
CIRCUIT_ADJ_SRV_TAB=P.AAC
.EXTRN NML$GB_EVTSRCTYP
.EXTRN NML$GQ_EVTSRCDS
.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$AL_ENTINF TAB
.EXTRN NML$AL_PERMINF TAB
.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_PRCODE, NML$GL_PRS_FLGS
.EXTRN NML$GL_NML_ENTITY
.EXTRN NML$GQ_NETNAMDSC
.EXTRN NML$GQ_RECBFDS
.EXTRN NML$GW_PRMDESCNT
.EXTRN LIB$ESTABLISH, LIB$REVERT
.EXTRN NML$OPENFILE, NML$BLD_REPLY
.EXTRN NML$MAINHANDLER
.EXTRN NML$ERROR 1, NML$ERROR 2
.EXTRN NML$LISTENTITY, NML$LIST_KNOWN_NODES
.EXTRN NML$LIST_TYPE_NODES
.EXTRN NML$LISTRKNOWN, NML$READACTLOG
.EXTRN NML$READKNOLOG, NML$READLOGGING
.EXTRN NML$SEND, NML$SHOWMULTIPLE
.EXTRN NML$SHOW_MULTIPLE_NODES
.EXTRN NML$SHOW_KNOWN_LOOP
.EXTRN NML$SHOWENTITY, NML$SHOW_CIRCUIT
.EXTRN NML$SHOWNODEBYNAME
.EXTRN NML$SHOWEXECUTOR
.EXTRN NML$CALL_NI_CONFIG

```

.PSECT \$CODE\$,NOWRT,2

		03FC 00000	.ENTRY	NML\$READ, Save R2,R3,R4,R5,R6,R7,R8,R9	: 0344
59	00000000G	00 9E 00002	MOVAB	NML\$GL_PRS_FLGS, R9	:
58	00000000'	00 9E 00009	MOVAB	TABLE TAB, R8	:
57	00000000G	00 9E 00010	MOVAB	NML\$GB_OPTIONS, R7	:
50	00000000G	00 9A 00017	MOVZBL	NML\$GB_INFO, R0	: 0379
		04 12 0001E	BNEQ	1\$: 0381
		54 D4 00020	CLRL	INDEX	: 0382
		2F 11 00022	BRB	6\$:
01		50 91 00024 1\$:	CMPB	R0, #1	: 0383
		05 12 00027	BNEQ	2\$:
54		01 D0 00029	MOVL	#1, INDEX	: 0384
		25 11 0002C	BRB	6\$:
02		50 91 0002E 2\$:	CMPB	R0, #2	: 0385
		05 12 00031	BNEQ	3\$:
54		02 D0 00033	MOVL	#2, INDEX	: 0386
		1B 11 00036	BRB	6\$:
03		50 91 00038 3\$:	CMPB	R0, #3	: 0387
		09 12 0003B	BNEQ	4\$:
		67 95 0003D	TSTB	NML\$GB_OPTIONS	: 0391
		0F 19 0003F	BLSS	5\$:
54		03 D0 00041	MOVL	#3, INDEX	: 0394
		0D 11 00044	BRB	6\$: 0391
04		50 91 00046 4\$:	CMPB	R0, #4	: 0395
		05 12 00049	BNEQ	5\$:
54		04 D0 0004B	MOVL	#4, INDEX	: 0396
		03 11 0004E	BRB	6\$:
54		01 CE 00050 5\$:	MNEGL	#1, INDEX	: 0398
FFFFFFF	8F	54 D1 00053 6\$:	CMPL	INDEX, #-1	: 0403

		0A	12	0005A	BNEQ	7\$				
		01	CE	0005C	MNEGL	#1, -(SP)		0404		
	00000000G	00	01	FB	0005F	CALLS	#1, NML\$ERROR_1			
		55	00	D0	00066	7\$:	MOVL	NML\$GL_NML_ENTITY, R5	0410	
		50	68	9E	0006D	MOVAB	TABLE_TAB, R0			
51		50	68	C1	00070	ADDL3	TABLE_TAB[R5], R0, ENT_TAB			
		53	00	98	00075	CVTBL	NML\$GB_ENTITY_FORMAT, R3	0442		
			51	D5	0007C	TSTL	ENT_TAB	0411		
			7A	13	0007E	BEQL	18\$			
56		61	51	C1	00080	ADDL3	ENT_TAB, (ENT_TAB), RTN_ADDR	0413		
		51	56	D1	00084	CMPL	RTN_ADDR, ENT_TAB	0419		
			71	13	00087	BEQL	18\$			
			67	95	00089	TSTB	NML\$GB_OPTIONS	0426		
			06	18	0008B	BGEQ	8\$			
		50	18	A1	9E	0008D	MOVAB	24(R1), READ_TABLE_ADR	0427	
			04	11	00091	BRB	9\$			
		50	04	A1	9E	00093	8\$:	MOVAB	4(R1), READ_TABLE_ADR	0429
	FFFFFFFC	8F	53	D1	00097	9\$:	CMPL	R3, #-4	0444	
			09	1F	0009E	BLSSU	10\$			
	FFFFFFFD	8F	53	D1	000A0	CMPL	R3, #-3			
			09	1B	000A7	BLEQU	11\$			
	FFFFFFFE	8F	53	D1	000A9	10\$:	CMPL	R3, #-1		
			10	12	000B0	BNEQ	13\$			
06		69	02	E1	000B2	11\$:	BBC	#2, NML\$GL_PRS_FLGS, 12\$	0447	
		52	0C	A0	D0	000B6	MOVL	12(READ_TABLE_ADR), READ_RTN	0448	
			2F	11	000BA	BRB	17\$			
		52	08	A0	D0	000BC	12\$:	MOVL	8(READ_TABLE_ADR), READ_RTN	0450
			29	11	000C0	BRB	17\$		0447	
	FFFFFFFE	8F	53	D1	000C2	13\$:	CMPL	R3, #-2	0451	
			06	12	000C9	BNEQ	14\$			
		52	10	A0	D0	000CB	MOVL	16(READ_TABLE_ADR), READ_RTN	0452	
			1A	11	000CF	BRB	17\$			
			53	D5	000D1	14\$:	TSTL	R3	0453	
			0F	13	000D3	BEQL	15\$			
		10	53	D1	000D5	CMPL	R3, #16		0456	
			0F	1A	000D8	BGTRU	16\$			
06		69	02	E1	000DA	BBC	#2, NML\$GL_PRS_FLGS, 15\$		0457	
		52	04	A0	D0	000DE	MOVL	4(READ_TABLE_ADR), READ_RTN	0458	
			07	11	000E2	BRB	17\$			
		52	60	D0	000E4	15\$:	MOVL	(READ_TABLE_ADR), READ_RTN	0460	
			02	11	000E7	BRB	17\$		0457	
			52	D4	000E9	16\$:	CLRL	READ_RTN	0462	
			0D	13	000EB	17\$:	BEQL	18\$	0468	
		52	51	C0	000ED	ADDL2	ENT_TAB, READ_RTN		0470	
			52	DD	000F0	PUSHL	READ_RTN		0476	
			54	DD	000F2	PUSHL	INDEX		0475	
			55	DD	000F4	PUSHL	R5		0474	
		66	03	FB	000F6	CALLS	#3, (RTN_ADDR)			
				04	000F9	RET			0468	
			53	DD	000FA	18\$:	PUSHL	R3	0485	
		7E	09	CE	000FC	MNEGL	#9, -(SP)			
	00000000G	00	02	FB	000FF	CALLS	#2, NML\$ERROR_2			
			04	0C	106	RET			0486	

; Routine Size: 263 bytes. Routine Base: \$CODE\$ + 0000

```

: 491 0487 1 %SBTTL 'NML_READ Read parameters'
: 492 0488 1 ROUTINE NML_READ (ENTITY, INDEX, READ_RTN) : NOVALUE =
: 493 0489 1
: 494 0490 1 !++
: 495 0491 1 FUNCTIONAL DESCRIPTION:
: 496 0492 1
: 497 0493 1 This routine dispatches to the routine which read the permanent
: 498 0494 1 or volatile data base for the specified entity.
: 499 0495 1
: 500 0496 1 INPUTS:
: 501 0497 1 ENTITY The internal NML index for the entity specified in
: 502 0498 1 the NICE command.
: 503 0499 1 INDEX Entity information table (in NMLDAT) index code.
: 504 0500 1 READ_RTN Address of routine to perform read requested
: 505 0501 1 by NICE message.
: 506 0502 1
: 507 0503 1 IMPLICIT INPUTS:
: 508 0504 1
: 509 0505 1 NML$GB_ENTITY_FORMAT contains the entity format code.
: 510 0506 1
: 511 0507 1 !--
: 512 0508 1
: 513 0509 2 BEGIN
: 514 0510 2
: 515 0511 2 MAP
: 516 0512 2 NML$GB_ENTITY_FORMAT : BYTE SIGNED;
: 517 0513 2
: 518 0514 2 SELECTONEU .NML$GB_ENTITY_FORMAT OF
: 519 0515 2 SET
: 520 0516 2 [NMASC_ENT_KNO, ! Known entities
: 521 0517 2 NMASC_ENT_ACT]: ! or active entities
: 522 0518 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
: 523 0519 2 .NML$GB_ENTITY_FORMAT, 0,
: 524 0520 2 .NML$GL_QUALIFIER_PST,
: 525 0521 2 .NML$GB_QUALIFIER_FORMAT,
: 526 0522 2 NML$AB_QUALIFIER_ID);
: 527 0523 2
: 528 0524 2 [1 TO 31]: ! Single entity
: 529 0525 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
: 530 0526 2 .NML$GB_ENTITY_FORMAT, NML$AB_ENTITY_ID,
: 531 0527 2 .NML$GL_QUALIFIER_PST,
: 532 0528 2 .NML$GB_QUALIFIER_FORMAT,
: 533 0529 2 NML$AB_QUALIFIER_ID);
: 534 0530 2
: 535 0531 2 [OTHERWISE]:
: 536 0532 2 NML$ERROR_2 (NMASC_STS_IDE, .NML$GB_ENTITY_CODE);
: 537 0533 2 TES;
: 538 0534 1 END; ! End of NML_READ

```

```

001C 00000 NML_READ:
54 00000000G 00 9E 00002 .WORD Save R2,R3,R4 : 0488
53 00000000G 00 9E 00009 MOVAB NML$GL_QUALIFIER_PST, R4 :
MOVAB NML$GB_QUALIFIER_FORMAT, R3 :

```

	52	00000000G	00	9E	00010	MOVAB	NMLSAB_QUALIFIER_ID, R2		
	50	00000000G	00	98	00017	CVTBL	NMLSGB_ENTITY_FORMAT, R0	0514	
FE	8F		50	91	0001E	CMPB	R0, #-2	0516	
			0B	1F	C0022	BLSSU	1\$		
			52	DD	00024	PUSHL	R2	0518	
	7E		63	9A	00025	MOVZBL	NMLSGB_QUALIFIER_FORMAT, -(SP)	0521	
			64	DD	00029	PUSHL	NMLSGL_QUALIFIER_PST	0520	
			7E	D4	0002B	CLRL	-(SP)	0518	
			16	11	0002D	BRB	2\$	0519	
			50	D5	0002F	TSTL	R0	0524	
			25	13	00031	BEQL	3\$		
	1F		50	91	00033	CMPB	R0, #31		
			20	1A	00036	BGTRU	3\$		
			52	DD	00038	PUSHL	R2	0525	
	7E		63	9A	0003A	MOVZBL	NMLSGB_QUALIFIER_FORMAT, -(SP)	0528	
			64	DD	0003D	PUSHL	NMLSGL_QUALIFIER_PST	0527	
		00000000G	00	9F	0003F	PUSHAB	NMLSAB_ENTITY_ID	0525	
			50	DD	00045	PUSHL	R0	0526	
		08	AC	DD	00047	PUSHL	INDEX	0525	
		0C	AC	DD	0004A	PUSHL	READ_RTN		
		04	AC	DD	0004D	PUSHL	ENTITY		
	00000000V	00	08	FB	00050	CALLS	#8, NML_READ_PLURAL		
				04	00057	RET			
			00	9A	00058	MOVZBL	NMLSGB_ENTITY_CODE, -(SP)	0532	
	7E	00000000G	00	9A	00058	MOVZBL	NMLSGB_ENTITY_CODE, -(SP)	0532	
	7E		09	CE	0005F	MNEGL	#9, -(SP)		
	00000000G	00	02	FB	00062	CALLS	#2, NMLSERROR_2		
			04	00069	RET			0534	

: Routine Size: 106 bytes, Routine Base: \$CODE\$ + 0107

```

: 540 0535 1 %SBTTL 'NML_READ_LOGGING Read logging volatile parameters'
: 541 0536 1 ROUTINE NML_READ_LOGGING (ENTITY, INDEX, READ_RTN) : NOVALUE =
: 542 0537 1
: 543 0538 1
: 544 0539 1 ++
: 545 0540 1 FUNCTIONAL DESCRIPTION:
: 546 0541 1 This routine reads the specified logging parameters from the volatile
: 547 0542 1 or permanent data base.
: 548 0543 1
: 549 0544 1 FORMAL PARAMETERS:
: 550 0545 1
: 551 0546 1 INPUTS:
: 552 0547 1 ENTITY The internal NML index for the entity specified in
: 553 0548 1 the NICE command.
: 554 0549 1 INDEX Entity information table (in NMLDAT) index code.
: 555 0550 1 READ_RTN Address of routine to perform read requested
: 556 0551 1 by NICE message.
: 557 0552 1
: 558 0553 1 IMPLICIT INPUTS:
: 559 0554 1 NML$GB_ENTITY_FORMAT contains the entity format code.
: 560 0555 1
: 561 0556 1 --
: 562 0557 1
: 563 0558 2 BEGIN
: 564 0559 2
: 565 0560 2 MAP
: 566 0561 2 NML$GB_ENTITY_FORMAT : BYTE SIGNED;
: 567 0562 2
: 568 0563 2 LOCAL
: 569 0564 2 LEN,
: 570 0565 2 ENTITY_ID;
: 571 0566 2
: 572 0567 2 SELECTONEU .NML$GB_ENTITY_FORMAT OF
: 573 0568 2 SET
: 574 0569 2 [NMASC_ENT_KNO,
: 575 0570 2 NMASC_ENT_ACT]: : Known and active entities
: 576 0571 2 BEGIN
: 577 0572 2 LEN = 0;
: 578 0573 2 ENTITY_ID = 0;
: 579 0574 2 END;
: 580 0575 2
: 581 0576 2 [NMASC_SNK_CON, : Console
: 582 0577 2 NMASC_SNK_FIL, : File
: 583 0578 2 NMASC_SNK_MON]: : Monitor
: 584 0579 2 BEGIN
: 585 0580 2 LEN = .NML$GB_ENTITY_FORMAT;
: 586 0581 2 ENTITY_ID = 0;
: 587 0582 2 END;
: 588 0583 2
: 589 0584 2 [OTHERWISE]:
: 590 0585 2 NML$ERROR_2 (NMASC_STS_IDE, NMASC_ENT_LOG); ! Option error
: 591 0586 2 TES;
: 592 0587 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX, .LEN, .ENTITY_ID, 0);
: 593 0588 2
: 594 0589 1 END; : End of NML_SHOW_LOGGING

```

				000C 00000 NML_READ_LOGGING:				
			0C	98	00002	WORD	Save R2,R3	: 0536
FE	50	00000000G	5C	91	00009	CVTBL	NML\$GB ENTITY_FORMAT, R0	: 0567
	8F		04	1F	0000D	CMPB	R0, #-2	: 0569
			53	D4	0C00F	BLSSU	1\$	
			0C	11	00011	CLRL	LEN	: 0572
			50	D5	0C013	BRB	2\$: 0573
			0C	13	00015	TSTL	R0	: 0576
	03		50	91	00017	BEQL	3\$	
			07	1A	0001A	CMPB	R0, #3	
	53		50	D0	0001C	BGTRU	3\$	
			52	D4	0001F	MOVL	R0, LEN	: 0580
			0C	11	00021	CLRL	ENTITY_ID	: 0581
			02	DD	00023	BRB	4\$: 0567
	7E		09	CE	00025	PUSHL	#2	: 0585
00000000G	00		02	FB	00028	MNEGL	#9, -(SP)	
			7E	D4	0002F	CALLS	#2, NML\$ERROR_2	
			52	DD	00031	CLRL	-(SP)	: 0587
			53	DD	00033	PUSHL	ENTITY_ID	
		08	AC	DD	00035	PUSHL	LEN	
		0C	AC	DD	00038	PUSHL	INDEX	
		04	AC	DD	0003B	PUSHL	READ RTN	
00000000V	00		06	FB	0003E	PUSHL	ENTITY	
			04	00045	CALLS	#6, NML_READ_PLURAL		
					RET			: 0589

; Routine Size: 70 bytes, Routine Base: \$CODE\$ + 0171

```

: 596 0590 1 %SBTTL 'NML_READ_NODE Read node parameters'
: 597 0591 1 ROUTINE NML_READ_NODE (ENTITY, INDEX, READ_RTN) : NOVALUE =
: 598 0592 1
: 599 0593 1 !++
: 600 0594 1 FUNCTIONAL DESCRIPTION:
: 601 0595 1 This routine dispatches to the routine which reads the permanent
: 602 0596 1 or volatile data base for nodes.
: 603 0597 1
: 604 0598 1 INPUTS:
: 605 0599 1 ENTITY The internal NML index for the entity specified in
: 606 0600 1 the NICE command.
: 607 0601 1 INDEX Entity information table (in NMLDAT) index code.
: 608 0602 1 READ_RTN Address of routine to perform read requested
: 609 0603 1 by NICE message.
: 610 0604 1
: 611 0605 1 IMPLICIT INPUTS:
: 612 0606 1 NML$GB_ENTITY_FORMAT contains the entity format code.
: 613 0607 1
: 614 0608 1 !--
: 615 0609 1
: 616 0610 2 BEGIN
: 617 0611 2
: 618 0612 2 MAP
: 619 0613 2 NML$GB_ENTITY_FORMAT : BYTE SIGNED,
: 620 0614 2 NML$GB_OPTIONS : BBLOCK [1];
: 621 0615 2
: 622 0616 2 SELECTONEU .NML$GB_ENTITY_FORMAT OF
: 623 0617 2 SET
: 624 0618 2 [NMASC_ENT_LOO]: ! Loop nodes (SHOW only)
: 625 0619 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
: 626 0620 2 0, 0);
: 627 0621 2
: 628 0622 2 [NMASC_ENT_KNO, ! Known nodes
: 629 0623 2 NMASC_ENT_ACT, ! Active nodes
: 630 0624 2 NMASC_ENT_ADJ]: ! Adjacent nodes
: 631 0625 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
: 632 0626 2 .NML$GB_ENTITY_FORMAT, 0,
: 633 0627 2 .NML$GL_QUALIFIER_PST,
: 634 0628 2 .NML$GB_QUALIFIER_FORMAT,
: 635 0629 2 NML$AB_QUALIFIER_ID);
: 636 0630 2
: 637 0631 2 [NMASC_ENT_ADD]: ! Node is specified by address
: 638 0632 2 IF .NML$GB_OPTIONS [NMASC_OPT_PER] THEN
: 639 0633 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
: 640 0634 2 2, NML$AB_ENTITY_ID)
: 641 0635 2 ELSE
: 642 0636 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
: 643 0637 2 0, .NML$AB_ENTITY_ID);
: 644 0638 2
: 645 0639 2 [1 TO 6]:
: 646 0640 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
: 647 0641 2 .NML$GB_ENTITY_FORMAT,
: 648 0642 2 NML$AB_ENTITY_ID);
: 649 0643 2
: 650 0644 2 [OTHERWISE]:
: 651 0645 2 NML$ERROR_2 (NMASC_STS_IDE, NMASC_ENT_NOD); ! Option error
: 652 0646 2 TES;

```


: 653

0647 1 END;

! End of NML_READ_NODE

		001C 00000 NML_READ_NODE:				
				.WORD	Save R2,R3,R4	0591
	54	00000000V	00 9E 00002	MOVAB	NML_READ_PLURAL, R4	
	53	00000000G	00 9E 00009	MOVAB	NML\$AB_ENTITY_ID, R3	
	52	00000000G	00 98 00010	CVTBL	NML\$GB_ENTITY_FORMAT, R2	0616
FD	8F		52 91 00017	CMPB	R2, #-3	0618
			04 12 0001B	BNEQ	1\$	
			7E D4 0001D	CLRL	-(SP)	0619
			44 11 0001F	BRB	5\$	
FC	8F		52 91 00021	1\$: CMPB	R2, #-4	0622
			06 13 00025	BEQL	2\$	
FE	8F		52 91 00027	CMPB	R2, #-2	
			24 1F 0002B	BLSSU	3\$	
		00000000G	00 9F 0002D	2\$: PUSHAB	NML\$AB_QUALIFIER_ID	0625
	7E	00000000G	00 9A 00033	MOVZBL	NML\$GB_QUALIFIER_FORMAT, -(SP)	0628
		00000000G	00 DD 0003A	PUSHL	NML\$GL_QUALIFIER_PST	0627
			7E D4 00040	CLRL	-(SP)	0625
			52 DD 00042	PUSHL	R2	0626
		08	AC DD 00044	PUSHL	INDEX	0625
		0C	AC DD 00047	PUSHL	READ RTN	
		04	AC DD 0004A	PUSHL	ENTITY	
	64		08 FB 0004D	CALLS	#8, NML_READ_PLURAL	
			04 00050	RET		
			52 D5 00051	3\$: TSTL	R2	0631
			14 12 00053	BNEQ	6\$	
		00000000G	00 95 00055	TSTB	NML\$GB_OPTIONS	0632
			06 18 0005B	BGEQ	4\$	
			53 DD 0005D	PUSHL	R3	0633
			02 DD 0005F	PUSHL	#2	
			0D 11 00061	BRB	7\$	
			63 DD 00063	4\$: PUSHL	NML\$AB_ENTITY_ID	0637
			7E D4 00065	5\$: CLRL	-(SP)	0636
			07 11 00067	BRB	7\$	
	06		52 91 00069	6\$: CMPB	R2, #6	0639
			0F 1A 0006C	BGTRU	8\$	
			0C BB 0006E	PUSHR	#*M<R2,R3>	0641
		08	AC DD 00070	7\$: PUSHL	INDEX	0640
		0C	AC DD 00073	PUSHL	READ RTN	
		04	AC DD 00076	PUSHL	ENTITY	
	64		05 FB 00079	CALLS	#5, NML_READ_PLURAL	
			04 0007C	RET		
			7E D4 0007D	8\$: CLRL	-(SP)	0645
	7E		09 CE 0007F	MNEGL	#9, -(SF)	
	00000000G	00	02 FB 00082	CALLS	#2, NML\$ERROR_2	
			04 00089	RET		0647

: Routine Size: 138 bytes, Routine Base: \$CODE\$ + 01B7

```

: 655 0648 1 XSBTTL 'NML_READ_EXECUTOR Read executor parameters'
: 656 0649 1 ROUTINE NML_READ_EXECUTOR (ENTITY, INDEX, READ_RTN) : NOVALUE =
: 657 0650 1
: 658 0651 1 !++
: 659 0652 1 ! FUNCTIONAL DESCRIPTION:
: 660 0653 1 ! This routine dispatches to the routine which reads the permanent
: 661 0654 1 ! or volatile data base to get information about the executor node.
: 662 0655 1
: 663 0656 1 ! INPUTS:
: 664 0657 1 ! ENTITY The internal NML index for the entity specified in
: 665 0658 1 ! the NICE command.
: 666 0659 1 ! INDEX Entity information table (in NMLDAT) index code.
: 667 0660 1 ! READ_RTN Address of routine to perform read requested
: 668 0661 1 ! by NICE message.
: 669 0662 1
: 670 0663 1 ! IMPLICIT INPUTS:
: 671 0664 1 ! NML$GB_ENTITY_FORMAT contains the entity format code.
: 672 0665 1
: 673 0666 1 !--
: 674 0667 1
: 675 0668 2 BEGIN
: 676 0669 2
: 677 0670 2 MAP
: 678 0671 2 NML$GB_ENTITY_FORMAT : BYTE SIGNED;
: 679 0672 2
: 680 0673 2 SELECTONEU .NML$GB_ENTITY_FORMAT OF
: 681 0674 2 SET
: 682 0675 2 [NMA$C_ENT_ADD]: ! Node address
: 683 0676 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX);
: 684 0677 2
: 685 0678 2 [1 TO 6]: ! Node name
: 686 0679 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX);
: 687 0680 2
: 688 0681 2 [OTHERWISE]:
: 689 0682 2 NML$ERROR_2 (NMA$C_STS_IDE, NMA$C_ENT_NOD); ! Option error
: 690 0683 2
: 691 0684 2 TES;
: 692 0685 1 END; ! End of NML_READ_EXECUTOR

```

```

                                0000 00000 NML_READ_EXECUTOR:
                                .WORD Save nothing
                                50 00000000G 00 98 00002 CVTBL NML$GB_ENTITY_FORMAT, R0
                                05 13 00009 BEQL 1$
                                06 50 91 0000B CMPB R0, #6
                                11 1A 0000E BGTRU 2$
                                08 AC DD 00010 1$: PUSHL INDEX
                                0C AC DD 00013 PUSHL READ_RTN
                                04 AC DD 00016 PUSHL ENTITY
                                00000000V 00 03 FB 00019 CALLS #3, NML_READ_PLURAL
                                04 04 00020 RET
                                7E D4 00021 2$: CLRL -(SP)
                                00000000G 7E 09 CE 00023 MNEGL #9, -(SP)
                                00 02 FB 00026 CALLS #2, NML$ERROR_2

```

NMLSREAD
V04-000

NML Read information module
NML_READ_EXECUTOR Read executor parameters

1 2
16-Sep-1984 00:28:30
14-Sep-1984 12:50:17

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[NML.SRC]NMLREAD.B32;1 Page 25
(8)

04 00020 RET

; 0685

; Routine Size: 46 bytes, Routine Base: \$CODE\$ + 0241

NM
VC

.....

```

0686 1 %SBTTL 'NML_READ_NETWORK Read X25-Protocol network parameters'
0687 1 ROUTINE NML_READ_NETWORK (ENTITY, INDEX, READ_RTN) : NOVALUF =
0688 1
0689 1 !++
0690 1 FUNCTIONAL DESCRIPTION:
0691 1 This routine dispatches to the routine which reads the permanent
0692 1 or volatile data base for X25-Protocol networks.
0693 1
0694 1 INPUTS:
0695 1 ENTITY The internal NML index for the entity specified in
0696 1 the NICE command.
0697 1 INDEX Entity information table (in NMLDAT) index code.
0698 1 READ_RTN Address of routine to perform read requested
0699 1 by NICE message.
0700 1
0701 1 IMPLICIT INPUTS:
0702 1 NML$GB_ENTITY_FORMAT contains the entity format code.
0703 1
0704 1 !--
0705 1
0706 2 BEGIN
0707 2
0708 2 MAP
0709 2 NML$GB_ENTITY_FORMAT : BYTE SIGNED;
0710 2
0711 2 SELECTONEU .NML$GB_ENTITY_FORMAT OF
0712 2 SET
0713 2 [0]: ! Active network (internal NML-ACP format only)
0714 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
0715 2 0, 0,
0716 2 0, 0, 0);
0717 2
0718 2 [NMASC_ENT_KNO]: ! Known entities
0719 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
0720 2 .NML$GB_ENTITY_FORMAT, 0,
0721 2 0, 0, 0);
0722 2
0723 2 [1 TO 16]: ! Single network
0724 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
0725 2 .NML$GB_ENTITY_FORMAT, NML$AB_ENTITY_ID,
0726 2 0, 0, 0);
0727 2
0728 2 [OTHERWISE]:
0729 2 NML$ERROR_2 (NMASC_STS_IDE, .NML$GB_ENTITY_CODE);
0730 2 TES;
0731 1 END; ! End of NML_READ_NETWORK

```

```

0000 0000 NML_READ_NETWORK:
50 0000000G 00 98 00002 .WORD Save nothing : 0687
08 12 00009 CVI BL NML$GB_ENTITY_FORMAT, R0 : 0711
7E 7C 0000B BNEQ 1$ : 0713
7E 7C 0000D CLRQ -(SP) : 0714

```

NMLSREAD
V04-000

NML Read information module
NML_READ_NETWORK Read X25-Protocol network par

K 2
16-Sep-1984 00:28:30
14-Sep-1984 12:50:17

VAX-11 Bliss-32 V4.0-742
DISK\$VM\$MASTER:[NML.SRC]NMLREAD.B32;1

Page 27
(9)

		7E	D4	0000F		CLRL	-(SP)			
		21	11	00011		BRB	4\$			
	FF	50	91	00013	1\$:	CMPB	R0, #-1		0718	
		06	12	00017		BNEQ	2\$			
		7E	7C	00019		CLRQ	-(SP)		0719	
		7E	7C	0001B		CLRQ	-(SP)			
		13	11	0001D		BRB	3\$		0720	
		50	D5	0001F	2\$:	TSTL	R0		0723	
		22	13	00021		BEQL	5\$			
		50	91	00023		CMPB	R0, #16			
	10	1D	1A	00026		BGTRU	5\$			
		7E	7C	00028		CLRQ	-(SP)		0724	
		7E	D4	0002A		CLRL	-(SP)			
		00	9F	0002C		PUSHAB	NML\$AB_ENTITY_ID			
		50	DD	00032	3\$:	PUSHL	R0		0725	
		08	AC	DD	4\$:	PUSHL	INDEX		0724	
		0C	AC	DD		PUSHL	READ RTN			
		04	AC	DD		PUSHL	ENTITY			
	00000000V	00	08	FB		CALLS	#8, NML_READ_PLURAL			
			04	00044		RET				
		7E	00000000G	00	9A	00045	5\$:	MOVZBL	NML\$GB_ENTITY_CODE, -(SP)	0729
		7E		09	CE	0004C		MNEGL	#9, -(SP)	
	00000000G	00	02	FB		0004F		CALLS	#2, NML\$ERROR_2	
			04	00056		RET			0731	

; Routine Size: 87 bytes, Routine Base: \$CODE\$ + 026F

```

741 0732 1 XSBTTL 'NML_READ_AREA Read area parameters'
742 0733 1 ROUTINE NML_READ_AREA (ENTITY, INDEX, READ_RTN) : NOVALUE =
743 0734 1
744 0735 1 |++
745 0736 1 | FUNCTIONAL DESCRIPTION:
746 0737 1 |
747 0738 1 | This routine dispatches to the routine which read the
748 0739 1 | volatile data base for areas.
749 0740 1 |
750 0741 1 | INPUTS:
751 0742 1 | ENTITY The internal NML index for the entity specified in
752 0743 1 | the NICE command: NML$C_AREA
753 0744 1 | INDEX Entity information table (in NMLDAT) index code.
754 0745 1 | READ_RTN Address of routine to perform read requested
755 0746 1 | by NICE message.
756 0747 1 |
757 0748 1 | IMPLICIT INPUTS:
758 0749 1 | NML$GB_ENTITY_FORMAT contains the entity format code.
759 0750 1 |
760 0751 1 | --
761 0752 1 |
762 0753 2 BEGIN
763 0754 2
764 0755 2 MAP
765 0756 2 NML$GB_ENTITY_FORMAT : BYTE SIGNED;
766 0757 2
767 0758 2 SELECTONEU .NML$GB_ENTITY_FORMAT OF
768 0759 2 SET
769 0760 2 [NMASC_ENT_KNO, ! Known entities
770 0761 2 NMASC_ENT_ACT]: ! or active entities
771 0762 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
772 0763 2 .NML$GB_ENTITY_FORMAT, 0,
773 0764 2 0, 0, 0); ! No qualifiers for areas
774 0765 2
775 0766 2 [0]: ! Single entity
776 0767 2 NML_READ_PLURAL (.ENTITY, .READ_RTN, .INDEX,
777 0768 2 0, .NML$GB_ENTITY_ID,
778 0769 2 0, 0, 0); ! No qualifiers for areas
779 0770 2
780 0771 2 [OTHERWISE]:
781 0772 2 NML$ERROR_2 (NMASC_STS_IDE, .NML$GB_ENTITY_CODE);
782 0773 2 TES;
783 0774 1 END: ! End of NML_READ_AREA

```

		0000 00000	NML_READ_AREA:			
				WORD	Save nothing	: 0733
				CVTBL	NML\$GB_ENTITY_FORMAT, R0	: 0758
FE	50	00000000G	00 98 00002	CMPB	R0, #-2	: 0760
	8F		50 91 00009	BLSSU	1\$	
			08 1F 0000D	CLRQ	-(SP)	: 0762
			7E 7C 0000F	CLRQ	-(SP)	
			7E 7C 00011	PUSHL	R0	: 0763
			50 DD 00013	BRB	2\$: 0762
			10 11 00015			

NML\$READ
V04-000

NML Read information module
NML_READ_AREA Read area parameters

M 2
16-Sep-1984 00:28:30
14-Sep-1984 12:50:17

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[NML.SRC]NMLREAD.B32;1
Page 29
(10)

		50	D5	00017	1\$:	TSTL	R0	:	0766	
		1D	12	00019		BNEQ	3\$:		
		7E	7C	0001B		CLRQ	-(SP)	:	0767	
		7E	D4	0001D		CLRL	-(SP)	:		
	00000000G	00	DD	0001F		PUSHL	NML\$AB_ENTITY_ID	:	0768	
		7E	D4	00025		CLRL	-(SP)	:	0767	
	08	AC	DD	00027	2\$:	PUSHL	INDEX	:		
	0C	AC	DD	0002A		PUSHL	READ RTN	:		
	04	AC	DD	0002D		PUSHL	ENTITY	:		
00000000V	00	08	FB	00030		CALLS	#8, NML_READ_PLURAL	:		
		04	00037			RET		:		
	7E	00000000G	00	9A	00038	3\$:	MOVZBL	NML\$GB_ENTITY_CODE, -(SP)	:	0772
	7E		09	CE	0003F		MNEGL	#9, -(SP)	:	
00000000G	00	02	FB	00042		CALLS	#2, NML\$ERROR_2	:		
		04	00049			RET		:	0774	

: Routine Size: 74 bytes, Routine Base: \$CODE\$ + 02C6

```

: 785 0775 1 %SBTTL 'NML_SHOW_LINKS Show link volatile parameters'
: 786 0776 1 ROUTINE NML_SHOW_LINKS (ENTITY, INDEX, READ_RTN) : NOVALUE =
: 787 0777 1
: 788 0778 1 !++
: 789 0779 1 ! FUNCTIONAL DESCRIPTION:
: 790 0780 1
: 791 0781 1 ! This routine shows the specified link parameters in the volatile
: 792 0782 1 ! data base.
: 793 0783 1
: 794 0784 1 ! INPUTS:
: 795 0785 1 ! ENTITY The internal NML index for the entity specified in
: 796 0786 1 ! the NICE command.
: 797 0787 1 ! INDEX Entity information table (in NMLDAT) index code.
: 798 0788 1 ! READ_RTN Address of routine to perform read requested
: 799 0789 1 ! by NICE message.
: 800 0790 1
: 801 0791 1 ! IMPLICIT INPUTS:
: 802 0792 1
: 803 0793 1 ! NML$GL_PRS_FLGS contains the parse flags.
: 804 0794 1 ! NML$GB_ENTITY_FORMAT contains the entity format code.
: 805 0795 1
: 806 0796 1 ! --
: 807 0797 1
: 808 0798 2 BEGIN
: 809 0799 2
: 810 0800 2 MAP
: 811 0801 2 NML$GB_ENTITY_FORMAT : BYTE SIGNED;
: 812 0802 2
: 813 0803 2 !
: 814 0804 2 ! All functions specifying the link entity must be system-specific.
: 815 0805 2 !
: 816 0806 2
: 817 0807 2 IF .NML$GL_PRS_FLGS [NML$V_PRS_VMS] THEN
: 818 0808 2 SELECT ONEU .NML$GB_ENTITY_FORMAT OF
: 819 0809 2 SET
: 820 0810 2 [NMA$C_ENT_KNO]: ! Known
: 821 0811 2 NML_READ_PLURAL (.ENTITY,
: 822 0812 2 .READ_RTN,
: 823 0813 2 .INDEX,
: 824 0814 2 NMA$C_ENT_KNO, 0,
: 825 0815 2 .NML$GL_QUALIFIER_PST,
: 826 0816 2 .NML$GB_QUALIFIER_FORMAT,
: 827 0817 2 NML$AB_QUALIFIER_ID);
: 828 0818 2
: 829 0819 2 [NMA$C_ENT_ADD]:
: 830 0820 2 NML_READ_PLURAL (.ENTITY,
: 831 0821 2 .READ_RTN,
: 832 0822 2 .INDEX,
: 833 0823 2 0,
: 834 0824 2 .(NML$AB_ENTITY_ID)<0,16>);
: 835 0825 2 TES;
: 836 0826 2
: 837 0827 2 NML$ERROR_2 (NMA$C_STS_IDE, NMA$C_SENT_LNK); ! Option error
: 838 0828 2
: 839 0829 1 END; ! End of NML_SHOW_LINKS

```


0004 00000 NML_SHOW_LINKS:						
				WORD	Save R2	: 0776
52	00000000V	00	9E 00002	MOVAB	NML_READ_PLURAL, R2	
4C	00000000G	00	E9 00009	BLBC	NML\$GL_PRS_FLGS, 2\$: 0807
50	00000000G	00	98 00010	CVTBL	NML\$GB_ENTITY_FORMAT, R0	: 0808
FF	8F	50	91 00017	CMPB	R0, #-T	: 0810
		26	12 0001B	BNEQ	1\$	
	00000000G	00	9F 0001D	PUSHAB	NML\$AB_QUALIFIER_ID	: 0811
7E	00000000G	00	9A 00023	MOVZBL	NML\$GB_QUALIFIER_FORMAT, -(SP)	: 0816
	00000000G	00	DD 0002A	PUSHL	NML\$GL_QUALIFIER_PST	: 0815
		7E	D4 00030	CLRL	-(SP)	: 0811
7E		01	CE 00032	MNEGL	#1, -(SP)	
	08	AC	DD 00035	PUSHL	INDEX	: 0813
	0C	AC	DD 00038	PUSHL	READ RTN	: 0812
	04	AC	DD 0003B	PUSHL	ENTITY	: 0811
62		08	FB 0003E	CALLS	#8, NML_READ_PLURAL	
		19	11 00041	BRB	2\$	
		50	D5 00043	TSTL	R0	: 0819
		15	12 00045	BNEQ	2\$	
7E	00000000G	00	3C 00047	MOVZWL	NML\$AB_ENTITY_ID, -(SP)	: 0824
		7E	D4 0004E	CLRL	-(SP)	: 0820
	08	AC	DD 00050	PUSHL	INDEX	: 0822
	0C	AC	DD 00053	PUSHL	READ RTN	: 0821
	04	AC	DD 00056	PUSHL	ENTITY	: 0820
62		05	FB 00059	CALLS	#5, NML_READ_PLURAL	
		07	DD 0005C	PUSHL	#7	: 0827
	7E	09	CE 0005E	MNEGL	#9, -(SP)	
00000000G	00	02	FB 00061	CALLS	#2, NML\$ERROR_2	
		04	00068	RET		: 0829

; Routine Size: 105 bytes. Routine Base: \$CODE\$ + 0310

```

841 0830 1 %SBTTL 'NML_READ_PLURAL Read plural entity parameters'
842 0831 1 ROUTINE NML_READ_PLURAL (ENT, RTN, INF, PRM2, PRM3, PRM4, PRM5, PRM6) : NOVALUE =
843 0832 1
844 0833 1 !++
845 0834 1 ! FUNCTIONAL DESCRIPTION:
846 0835 1
847 0836 1 ! This routine performs initialization for read operations.
848 0837 1 ! The NICE framing messages (plural and done) are transmitted and
849 0838 1 ! the SHOW or LIST routine is called.
850 0839 1
851 0840 1 ! FORMAL PARAMETERS:
852 0841 1
853 0842 1 ! ENT Entity type code.
854 0843 1 ! RTN Address of routine to be called.
855 0844 1 ! INF Information code (SUMMARY, STATUS, CHARACTERISTICS,
856 0845 1 ! or COUNTERS).
857 0846 1 ! PRM2 - PRM6 Function-specific routine parameters.
858 0847 1
859 0848 1 ! SIDE EFFECTS:
860 0849 1
861 0850 1 ! Several NICE messages are transmitted.
862 0851 1
863 0852 1 !--
864 0853 1
865 0854 2 BEGIN
866 0855 2
867 0856 2 MAP
868 0857 2 NML$GB_OPTIONS : BBLOCK [1];
869 0858 2
870 0859 2 LOCAL
871 0860 2 MSGSIZE;
872 0861 2
873 0862 2 IF .NML$GB_OPTIONS [NMA$V_OPT_PER] THEN
874 0863 2 !
875 0864 2 ! Open permanent data base file specified for write.
876 0865 2 !
877 0866 2 NML$OPENFILE (.NML$AB_ENTITYDATA [.ENT, EIT$B_FILEID], NMA$C_OPN_AC_RO);
878 0867 2 !
879 0868 2 ! Send success with multiple responses message.
880 0869 2
881 0870 2 NML$BLD_REPLY (UPLIT (0, NMA$C_STS_MOR), MSGSIZE); ! Build message
882 0871 2 NML$SEND (NML$AB_SNDBUFFER, .MSGSIZE); ! Send it
883 0872 2
884 0873 2 ! Enable condition handler to allow done message to be sent.
885 0874 2
886 0875 2 LIB$ESTABLISH (NML$MAINHANDLER);
887 0876 2
888 0877 2 ! Call entity-specific routine.
889 0878 2
890 0879 2 (.RTN) (.ENT, .INF, .PRM2, .PRM3, .PRM4, .PRM5, PRM6);
891 0880 2
892 0881 2 ! Signal done message.
893 0882 2
894 0883 2 LIB$REVERT (); ! Disable condition handler
895 0884 2 NML$ERROR_1 (NMA$C_STS_DON); ! Signal no more responses
896 0885 2
897 0886 1 END; ! End of NML_READ_PLURAL

```

```

.PSECT $PLITS$,NOWRT,NOEXE,2
00000002 00000000 0000C P.AAD: .LONG 0, 2 ;

.PSECT $CODES$,NOWRT,2
0000 00000 NML_READ_PLURAL:
:WORD Save nothing ; 0831
SUBL2 #4, SP ;
TSTB NML$GB_OPTIONS ; 0862
BGEQ 1$ ;
CLRL -(SP) ; 0866
MULL3 #44, ENT, R0 ;
MOVZBL NML$AB_ENTITYDATA[R0], -(SP) ;
CALLS #2, NML$OPENFILE ;
PUSHL SP ; 0870
PUSHAB P.AAD ;
CALLS #2, NML$BLD_REPLY ;
PUSHL MSGSIZE ; 0871
PUSHAB NML$AB_SNDBUFFER ;
CALLS #2, NML$SEND ;
PUSHAB NML$MAINHANDLER ; 0875
CALLS #1, LIB$ESTABLISH ;
PUSHAB PRM6 ; 0879
MOVQ PRM4, -(SP) ;
MOVQ PRM2, -(SP) ;
PUSHL INF ;
PUSHL ENT ;
CALLS #7, @RTN ;
CALLS #0, LIB$REVERT ; 0883
CVTBL #-128, -(SP) ; 0884
CALLS #1, NML$ERROR_1 ;
RET ; 0886

```

; Routine Size: 118 bytes, Routine Base: \$CODES + 0379

NML\$READ
V04-000

NML Read information module
NML_READ_PLURAL Read plural entity parameters

E 3
16-Sep-1984 00:28:30
14-Sep-1984 12:50:17

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[NML.SRC]NMLREAD.B32;1 (13) Page 34

: 899 0887 1 END
: 900 0888 1
: 901 0889 0 ELUDOM

: End of module

PSECT SUMMARY

Name	Bytes	Attributes
\$OWNS	984	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$SPLITS	20	NOVEC, NOWRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$CODES	1007	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

Library Statistics

File	Total	Symbols Loaded	Percent	Pages Mapped	Processing Time
-\$255\$DUA28:[NML.OBJ]NMLLIB.L32;1	341	33	9	27	00:00.1
-\$255\$DUA28:[SHRLIB]NMLIBRY.L32;1	887	22	2	47	00:00.2
-\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	0	0	581	00:02.2

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:NMLREAD/OBJ=OBJ\$:NMLREAD MSRCS\$:NMLREAD/UPDATE=(ENH\$:NMLREAD)

: Size: 1007 code + 1004 data bytes
: Run Time: 00:29.2
: Elapsed Time: 01:12.8
: Lines/CPU Min: 1827
: Lexemes/CPU-Min: 28538
: Memory Used: 171 pages
: Compilation Complete

The image displays a grid of 100 small terminal window screenshots, arranged in 10 rows and 10 columns. Each window shows a different view of system data or logs. Several windows are prominently labeled with titles:

- NMLPURGE LIS**: Located in the top right area.
- NMLPARINI LIS**: Located in the second row, second column.
- NMLNOOFTL LIS**: Located in the third row, first column.
- NMLREAD LIS**: Located in the fifth row, tenth column.
- NMLPARPRM LIS**: Located in the sixth row, fifth column.
- NMLPMANTP LIS**: Located in the eighth row, seventh column.

The text within the windows is mostly monospaced and appears to be system logs or diagnostic output. The overall appearance is that of a multi-processor system's monitoring interface.

