

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	NNNNNN	NNN	MMM	MMM	LLL
NNN	NNNNNN	NNN	MMM	MMM	LLL
NNN	NNNNNN	NNN	MMM	MMM	LLL
NNN	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	LLLLLLLLLLLLLLLL
NNN	NNN	NNN	MMM	MMM	LLLLLLLLLLLLLLLL
NNN	NNN	NNN	MMM	MMM	LLLLLLLLLLLLLLLL

_S

Ps

NP

NP

\$G

\$O

NP

PA

_L

```

LL          IIIII
LL          IIIII
        II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LLLLLLLLLLL
LLLLLLLLLLL
        IIIII
        IIIII
SSSSSSSSS
SSSSSSSSS
    SS
    SS
    SS
    SS
        SSSSSS
        SSSSSS
                SS
                SS
                SS
                SS
SSSSSSSSS
SSSSSSSSS

```

: 1


```
0001 0 %TITLE 'NML LIST permanent parameter module'
0002 0 MODULE NML$LIST (
0003 0     LANGUAGE (BLISS32),
0004 0     ADDRESSING_MODE (NONEXTERNAL=GENERAL),
0005 0     ADDRESSING_MODE (EXTERNAL=GENERAL),
0006 0     IDENT = 'V04-000'
0007 0 ) =
0008 1 BEGIN
0009 1
0010 1 *****
0011 1 *
0012 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0013 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0014 1 *  ALL RIGHTS RESERVED.
0015 1 *
0016 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0017 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0018 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0019 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0020 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0021 1 *  TRANSFERRED.
0022 1 *
0023 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0024 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0025 1 *  CORPORATION.
0026 1 *
0027 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0028 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0029 1 *
0030 1 *****
0031 1
0032 1
0033 1
0034 1 ++
0035 1 FACILITY: DECnet-VAX Network Management Listener
0036 1
0037 1 ABSTRACT:
0038 1
0039 1     This module contains routines for processing the NCP LIST command.
0040 1
0041 1 ENVIRONMENT: VAX/VMS Operating System
0042 1
0043 1 AUTHOR: Distributed Systems Software Engineering
0044 1
0045 1 CREATION DATE: 30-DEC-1979
0046 1
0047 1 MODIFIED BY:
0048 1
0049 1     V03-004 MKP0005      Kathy Perko      4-Aug-1983
0050 1     Enhance access time to the node permanent database by using
0051 1     multiple ISAM keys to speed up access.
0052 1
0053 1     V03-003 MKP0004      Kathy Perko      26-April-1983
0054 1     Make GETTABLE a global routine for SET MODULE CONFIGURATOR
0055 1     ALL command processing.
0056 1
0057 1     V03-002 MKP0003      Kathy Perko      5-July-1982
```

NML\$LIST
V04-000

NML LIST permanent parameter module

N 6
16-Sep-1984 00:18:35
14-Sep-1984 12:50:10

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[NML.SRC]NMLLIST.B32;1 Page 2 (1)

:	58	0058	1	:	
:	59	0059	1	:	
:	60	0060	1	:	
:	61	0061	1	:	
:	62	0062	1	:	
:	63	0063	1	:	
:	64	0064	1	:	
:	65	0065	1	:	
:	66	0066	1	:	
:	67	0067	1	:	
:	68	0068	1	:	--
:	69	0069	1	:	

Add support to list X25-Protocol Group entity (a single
entity show is essentially a multiple show command because
there can be more than one DTE per group.)
Add X29 Server and Trace Modules.

V03-001 MKP0002 Kathy Perko 16-June-1982
Add handling for entity qualifiers.

V02-001 MKP0001 Kathy Perko 23-July-1981
Add circuit entity and multidrop lines.


```

: 71 0070 1 %SBTTL 'Declarations'
: 72 0071 1
: 73 0072 1
: 74 0073 1 TABLE OF CONTENTS:
: 75 0074 1
: 76 0075 1
: 77 0076 1 FORWARD ROUTINE
: 78 0077 1 NML$LISTKNOWN : NOVALUE,
: 79 0078 1 NML$LIST_KNOWN_NODES: NOVALUE,
: 80 0079 1 NML$LISTENTITY : NOVALUE,
: 81 0080 1 NML_LISTENTITY,
: 82 0081 1 NML$LIST_TYPE_NODES: NOVALUE,
: 83 0082 1 NML$GETTABLE : NOVALUE;
: 84 0083 1
: 85 0084 1
: 86 0085 1 INCLUDE FILES:
: 87 0086 1
: 88 0087 1
: 89 0088 1 LIBRARY 'LIB$:NMLLIB.L32';
: 90 0089 1 LIBRARY 'SHRLIB$:NMLIBRY.L32';
: 91 0090 1 LIBRARY 'SYSS$LIBRARY:STARLET.L32';
: 92 0091 1
: 93 0092 1
: 94 0093 1 EQUATED SYMBOLS:
: 95 0094 1
: 96 0095 1
: 97 0096 1
: 98 0097 1 OWN STORAGE:
: 99 0098 1
: 100 0099 1
: 101 0100 1
: 102 0101 1 Entity buffer and descriptor.
: 103 0102 1
: 104 0103 1 OWN
: 105 0104 1 NML$T_ENTITYBUF : BBLOCK [NML$K_ENTBUFLN],
: 106 0105 1 NML$Q_ENTITYDSC : DESCRIPTOR;
: 107 0106 1
: 108 0107 1
: 109 0108 1 EXTERNAL REFERENCES:
: 110 0109 1
: 111 0110 1
: 112 0111 1 $NML_EXTDEF;
: 113 0112 1
: 114 0113 1 EXTERNAL ROUTINE
: 115 0114 1 NML$OPENFILE,
: 116 0115 1 NML$BLD_REPLY,
: 117 0116 1 NML$ERROR_1,
: 118 0117 1 NML$ERROR_2,
: 119 0118 1 NML$GETRECOWNER,
: 120 0119 1 NML$MATCHRECORD,
: 121 0120 1 NML$READPARLIST,
: 122 0121 1 NML$READRECORD,
: 123 0122 1 NML$READ_KNOWN_NODE_REC,
: 124 0123 1 NML$SEND;
: 125 0124 1
```



```
127 0125 1 %SBTTL 'NML$LISTKNOWN List known entity parameters'
128 0126 1 GLOBAL ROUTINE NML$LISTKNOWN (ENTITY, INF, ENTITY_LEN, ENTITY_ADR) : NOVALUE =
129 0127 1
130 0128 1 ++
131 0129 1 FUNCTIONAL DESCRIPTION:
132 0130 1
133 0131 1 This routine lists entries for all entities of the specified type
134 0132 1 in the permanent data base.
135 0133 1
136 0134 1 FORMAL PARAMETERS:
137 0135 1
138 0136 1 ENTITY Entity type code.
139 0137 1 INF Information type code.
140 0138 1 ENTITY_LEN Byte count of entity id string (used only if there
141 0139 1 is a qualifier in the command, in which case it is
142 0140 1 essentially a multiple LIST command.)
143 0141 1 ENTITY_ADR Address of entity id string (used only if there is
144 0142 1 a qualifier in the command, in which case it is
145 0143 1 essentially a multiple LIST command.)
146 0144 1
147 0145 1 --
148 0146 1
149 0147 2 BEGIN
150 0148 2
151 0149 2 LOCAL
152 0150 2 FID, ! File id code
153 0151 2 KEY, ! Temporary record key buffer
154 0152 2 OWNER, ! Data base search key
155 0153 2 RECDSC : DESCRIPTOR, ! Record descriptor
156 0154 2 TABDSC : REF DESCRIPTOR;
157 0155 2
158 0156 2
159 0157 2 Set up entity-specific information.
160 0158 2
161 0159 2 SELECTONEU .ENTITY OF
162 0160 2 SET
163 0161 2
164 0162 2 [NML$C_CIRCUIT]:
165 0163 2 BEGIN
166 0164 2
167 0165 2 Open node file to get loop node name.
168 0166 2 (If open fails, return other information anyway.)
169 0167 2
170 0168 2 NML$OPENFILE (NMASC_OPN_NODE, NMASC_OPN_AC_RO);
171 0169 2
172 0170 2 END;
173 0171 2
174 0172 2 TES;
175 0173 2
176 0174 2 Get entity information.
177 0175 2
178 0176 2 FID = .NML$AB_ENTITYDATA [.ENTITY, EIT$B_FILEID]; ! Get file id
179 0177 2 OWNER = .NML$AB_ENTITYDATA [.ENTITY, EIT$W_KEY]; ! Get search key
180 0178 2
181 0179 2 Get table.
182 0180 2
183 0181 2 NML$GETTABLE (.ENTITY, .INF, TABDSC);
```



```
184 0182 2 !
185 0183 2 ! Try to match record in file.
186 0184 2 !
187 0185 2 KEY = 0; ! Initialize record key
188 0186 2 WHILE NML$MATCHRECORD (.FID, NML$GQ_RECBUF, KEY, OWNER,
189 0187 2 .ENTITY_LEN, .ENTITY_ADR,
190 0188 2 0, 0, 0, RECDSC) DO
191 0189 2 BEGIN
192 0190 2 NML_LISTENTITY (.ENTITY, RECDSC, .TABDSC);
193 0191 2 KEY = .KEY + 1;
194 0192 2 END;
195 0193 2
196 0194 1 END; ! End of NML$LISTKNOWN
```

```
.TITLE NML$LIST NML LIST permanent parameter module
.IDENT \V04-000\
```

```
.PSECT $OWNS$,NOEXE,2
```

```
00000 NML$T_ENTITYBUF:
```

```
.BLKB 64
```

```
00040 NML$Q_ENTITYDSC:
```

```
.BLKB 8
```

```
.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$AL_ENTINF TAB
.EXTRN NML$AL_PERMINTAB
.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
```


				001C 00000	.ENTRY	NML\$LISTKNOWN, Save R2,R3,R4	: 0126
	5E		10	C2 00002	SUBL2	#16, SP	:
	52	04	AC	D0 00005	MOVL	ENTITY, R2	: 0159
	09		52	D1 00009	CMPL	R2, #9	: 0162
			09	12 0000C	BNEQ	1\$	:
			7E	7C 0000E	CLRQ	-(SP)	: 0168
	00000000G	00	02	FB 00010	CALLS	#2, NML\$OPENFILE	:
50			2C	C5 00017	MULL3	#4, R2, R0	: 0176
			04	9A 0001B	MOVZBL	NML\$AB_ENTITYDATA[R0], FID	:
	00000000G	0040	9F	00023	PUSHAB	NML\$AB_ENTITYDATA+3[R0]	: 0177
	00000000G	0040	9E	3C 0002A	MOVZWL	@(SP)+, OWNER	:
			5E	DD 0002D	PUSHL	SP	: 0181
		08	AC	DD 0002F	PUSHL	INF	:
			52	DD 00032	PUSHL	R2	:
	00000000V	00	03	FB 00034	CALLS	#3, NML\$GETTABLE	:
		04	AE	D4 0003B	CLRL	KEY	: 0185
		08	AE	9F 0003E	PUSHAB	RECDSC	: 0186
			7E	7C 00041	CLRQ	-(SP)	:
			7E	D4 00043	CLRL	-(SP)	:
		0C	AC	7D 00045	MOVQ	ENTITY_LEN, -(SP)	: 0187
			53	DD 00049	PUSHL	OWNER	: 0186
		20	AE	9F 0004B	PUSHAB	KEY	:
		00000000G	00	9F 0004E	PUSHAB	NML\$GQ_RECBFDSC	:
			54	DD 00054	PUSHL	FID	:
	00000000G	00	0A	FB 00056	CALLS	#10, NML\$MATCHRECORD	:
		13	50	E9 0005D	BLBC	R0, 3\$	:
			6E	DD 00060	PUSHL	TABDSC	: 0190
		0C	AE	9F 00062	PUSHAB	RECDSC	:
			52	DD 00065	PUSHL	R2	:
	00000000V	00	03	FB 00067	CALLS	#3, NML_LISTENTITY	:
		04	AE	D6 0006E	INCL	KEY	: 0191
			CB	11 00071	BRB	2\$	: 0186
			04	00073	RET		: 0194

; Routine Size: 116 bytes, Routine Base: \$CODE\$ + 0000

; 197 0195 1

NML\$LIST
V04-000

NML LIST permanent parameter module
NML\$LIST_KNOWN_NODES List known nodes

F 7
16-Sep-1984 00:18:35
14-Sep-1984 12:50:10

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[NML.SRC]NMLLIST.B32;1 Page 7
(4)

```

199 0196 1 %SBTTL 'NML$LIST_KNOWN_NODES List known nodes'
200 0197 1 GLOBAL ROUTINE NML$LIST_KNOWN_NODES (ENTITY, INF, DUM1, DUM2) : NOVALUE =
201 0198 1
202 0199 1 !++
203 0200 1 FUNCTIONAL DESCRIPTION:
204 0201 1
205 0202 1 This routine lists all entries for nodes that are in the permanent
206 0203 1 data base. The executor node is listed first. Remote nodes are
207 0204 1 then listed. Loop nodes are listed last.
208 0205 1
209 0206 1 FORMAL PARAMETERS:
210 0207 1
211 0208 1 ENTITY Entity type code.
212 0209 1 INF Information type code.
213 0210 1 DUM1 Not used.
214 0211 1 DUM2 Not used.
215 0212 1
216 0213 1 IMPLICIT INPUTS:
217 0214 1
218 0215 1 NONE
219 0216 1
220 0217 1 IMPLICIT OUTPUTS:
221 0218 1
222 0219 1 NONE
223 0220 1
224 0221 1 ROUTINE VALUE:
225 0222 1 COMPLETION CODES:
226 0223 1
227 0224 1 NONE
228 0225 1
229 0226 1 SIDE EFFECTS:
230 0227 1
231 0228 1 NONE
232 0229 1
233 0230 1 --
234 0231 1
235 0232 2 BEGIN
236 0233 2
237 0234 2 Return executor node.
238 0235 2
239 0236 2 nml$list_type_nodes (nml$sc_executor, .inf, 0, 0);
240 0237 2
241 0238 2 Return remote nodes.
242 0239 2
243 0240 2 nml$list_type_nodes (nml$sc_node, .inf, 0, 0);
244 0241 2
245 0242 2 Return loop nodes.
246 0243 2
247 0244 2 nml$list_type_nodes (nml$sc_loopnode, .inf, 0, 0);
248 0245 2
249 0246 1 END; ! End of NML$LIST_KNOWN_NODES
```

0004 00000

.ENTRY NML\$LIST_KNOWN_NODES, Save R2

: 0197

: 250 0247 1


```
: 252 0248 1 %SBTTL 'NML$LISTENTITY List entity parameters'
: 253 0249 1 GLOBAL ROUTINE NML$LISTENTITY (ENTITY, INF, ENTITY_LEN, ENTITY_ADR) : NOVALUE =
: 254 0250 1
: 255 0251 1 ++
: 256 0252 1 FUNCTIONAL DESCRIPTION:
: 257 0253 1
: 258 0254 1 This routine lists permanent data base information for the specified
: 259 0255 1 entity.
: 260 0256 1
: 261 0257 1 FORMAL PARAMETERS:
: 262 0258 1
: 263 0259 1 ENTITY Entity type code.
: 264 0260 1 INF Information type code.
: 265 0261 1 ENTITY_LEN Byte count of entity id string.
: 266 0262 1 ENTITY_ADR Address of entity id string.
: 267 0263 1
: 268 0264 1 --
: 269 0265 1
: 270 0266 2 BEGIN
: 271 0267 2
: 272 0268 2 LOCAL
: 273 0269 2 fid, ! File id code
: 274 0270 2 key, ! Temporary record key buffer
: 275 0271 2 key_value_dsc: VECTOR [2],
: 276 0272 2 detail, ! Error detail
: 277 0273 2 owner, ! Data base search key
: 278 0274 2 recdsc : DESCRIPTOR, ! Record descriptor
: 279 0275 2 tabdsc : REF DESCRIPTOR,
: 280 0276 2 status,
: 281 0277 2 node_type;
: 282 0278 2
: 283 0279 2
: 284 0280 2 If LISTing a circuit, open node file in case there's a loop node set up
: 285 0281 2 on the circuit. In that case I must get loop node name to add to the
: 286 0282 2 NCP response message. (If open fails, return other information anyway.)
: 287 0283 2
: 288 0284 2
: 289 0285 2 IF .entity EQL nml$sc_circuit THEN
: 290 0286 2 nml$openfile (nma$sc_opn_node, nma$sc_opn_ac_ro);
: 291 0287 2
: 292 0288 2 Get entity information.
: 293 0289 2
: 294 0290 2 fid = .nml$ab_entitydata [.entity, eit$b_fileid]; ! Get file id
: 295 0291 2 owner = .nml$ab_entitydata [.entity, eit$b_key]; ! Search key
: 296 0292 2 detail = .nml$ab_entitydata [.entity, eit$b_detail]; ! Error detail
: 297 0293 2
: 298 0294 2 Try to get the entity's record from the permanent database file.
: 299 0295 2 The node database organization is different from the others (it uses
: 300 0296 2 3 ISAM keys instead of one) to make it more efficient. This is
: 301 0297 2 necessary because of it's size.
: 302 0298 2
: 303 0299 2 node_type = 0;
: 304 0300 2 IF .fid EQL nma$sc_opn_node THEN
: 305 0301 2 BEGIN
: 306 0302 2 IF .entity EQL nml$sc_executor THEN
: 307 0303 2 BEGIN
: 308 0304 2 key_value_dsc [0] = nmn$sc_typ_key_len;
```

```

309 0305 4      key_value_dsc [1] = UPLIT (nml$c_executor);
310 0306 4      END
311 0307 3      ELSE
312 0308 4      BEGIN
313 0309 4      key_value_dsc [0] = .entity_len;
314 0310 4      key_value_dsc [1] = .entity_adr;
315 0311 4      END;
316 0312 4      status = nml$readrecord (.fid,      ! Node DB file ID
317 0313 4      owner,      ! ISAM key type to use.
318 0314 4      key_value_dsc,      ! Entity ID descriptor address
319 0315 4      nml$gq_recbfdsc,      ! Read buffer.
320 0316 4      recdsc,      ! Return data descriptor.
321 0317 4      node_type);      ! Not used for list.
322 0318 4      END
323 0319 4      ELSE
324 0320 4      BEGIN
325 0321 4      key = 0;      ! Initialize record key
326 0322 4      status = nml$matchrecord (.fid, nml$gq_recbfdsc, key, .owner,
327 0323 4      .entity_len, .entity_adr,
328 0324 4      0, 0, 0, recdsc);
329 0325 4      END;
330 0326 4      IF .status THEN
331 0327 4      BEGIN
332 0328 4      ! If the entity read was a node that turned out to be the executor,
333 0329 4      ! make sure the LIST information returned to NCP is for the executor.
334 0330 4      IF .node_type EQL nml$c_executor THEN
335 0331 4      entity = nml$c_executor;
336 0332 4      ! Get table of parameters to include in LIST type (STATUS, SUMMARY, or CHAR).
337 0333 4      nml$gettable (.entity, .inf, tabdsc);
338 0334 4      nml$listentity (.entity, recdsc, .tabdsc);
339 0335 4      END
340 0336 4      ELSE
341 0337 4      nml$error_2 (nma$c_sts_cmp, .detail); ! Signal error message
342 0338 4
343 0339 4      ! End of NML$LISTENTITY
344 0340 4      END;
345 0341 4
346 0342 4
347 0343 4
```

.PSECT \$PLITS\$,NOWRT,NOEXE,2

00000007 00000 P.AAA: .LONG 7

.PSECT \$CODE\$,NOWRT,2

```

54 00000000G 00 001C 00000
53 00000000G 00 9E 00002
5E 20 C2 00010
09 04 AC D1 00013
09 09 12 00017
7E 7C 00019
```

```

ENTRY NML$LISTENTITY, Save R2,R3,R4
MOVAB NML$GQ_RECBFDSC, R4
MOVAB NML$AB_ENTITYDATA, R3
SUBL2 #32, SP
CMPL ENTITY, #9
BNEQ 1$
CLRQ -(SP)
```

: 0249

: 0285

: 0286

50	00000000G	00	02	FB	0001B	CALLS	#2, NML\$OPENFILE	:	
	04	AC	2C	C5	00022	MULL3	#4, ENTITY, R0	:	0290
		51	6340	9A	00027	MOVZBL	NML\$AB_ENTITYDATA[R0], FID	:	
			03	A340	9F	PUSHAB	NML\$AB_ENTITYDATA+3[R0]	:	0291
	04	AE	9E	3C	0002F	MOVZWL	@(SP)+, OWNER	:	
			01	A340	9F	PUSHAB	NML\$AB_ENTITYDATA+1[R0]	:	0292
		52	9E	3C	00037	MOVZWL	@(SP)+, DETAIL	:	
			6E	D4	0003A	CLRL	NODE_TYPE	:	0299
			51	D5	0003C	TSTL	FID	:	0300
			31	12	0003E	BNEQ	4\$	:	
	07		04	AC	D1	CMPL	ENTITY, #7	:	0302
				0E	12	BNEQ	2\$	:	
18	AE		02	D0	00046	MOVL	#2, KEY_VALUE_DSC	:	0304
1C	AE	00000000'	00	9E	0004A	MOVAB	P.AAA, KEY_VALUE_DSC+4	:	0305
			05	11	00052	BRB	3\$	:	0302
18	AE		0C	AC	7D	MOVQ	ENTITY_LEN, KEY_VALUE_DSC	:	0309
				5E	DD	PUSHL	SP	:	0312
			14	AE	9F	PUSHAB	RECDSC	:	
				54	DD	PUSHL	R4	:	
			24	AE	9F	PUSHAB	KEY_VALUE_DSC	:	
			14	AE	9F	PUSHAB	OWNER	:	
				51	DD	PUSHL	FID	:	
00000000G	00		06	FB	00068	CALLS	#6, NML\$READRECORD	:	0300
			1D	11	0006F	BRB	5\$	:	0321
			08	AE	D4	CLRL	KEY	:	0322
			10	AE	9F	PUSHAB	RECDSC	:	
				7E	7C	CLRQ	-(SP)	:	
				7E	D4	CLRL	-(SP)	:	
	7E		0C	AC	7D	MOVQ	ENTITY_LEN, -(SP)	:	0323
			1C	AE	DD	PUSHL	OWNER	:	0322
			24	AE	9F	PUSHAB	KEY	:	
				12	BB	PUSHR	#*M<R1,R4>	:	
00000000G	00		0A	FB	00087	CALLS	#10, NML\$MATCHRECORD	:	
	28		50	E9	0008E	BLBC	STATUS, 7\$	:	0326
	07		6E	D1	00091	CMPL	NODE_TYPE, #7	:	0332
			04	12	00094	BNEQ	6\$	:	
04	AC		07	D0	00096	MOVL	#7, ENTITY	:	0333
			0C	AE	9F	PUSHAB	TABDSC	:	0337
	7E		04	AC	7D	MOVQ	ENTITY, -(SP)	:	
00000000V	00		03	FB	000A1	CALLS	#3, NML\$GETTABLE	:	
			0C	AE	DD	PUSHL	TABDSC	:	0338
			14	AE	9F	PUSHAB	RECDSC	:	
			04	AC	DD	PUSHL	ENTITY	:	
00000000V	00		03	FB	000B1	CALLS	#3, NML_LISTENTITY	:	
				04	000B8	RET		:	0326
			52	DD	000B9	PUSHL	DETAIL	:	0341
	7E		08	C:	000BB	MNEGL	#8, -(SP)	:	
00000000G	00		02	FB	000BE	CALLS	#2, NML\$ERROR_2	:	
			04	000C5	RET			:	0343

; Routine Size: 198 bytes, Routine Base: \$CODE\$ + 009C

```

349 0344 1 %SBTTL 'NML_LISTENTITY List known entity parameters'
350 0345 1 ROUTINE NML_LISTENTITY (ENTITY, RECDSC, TABDSC) =
351 0346 1
352 0347 1 ++
353 0348 1 FUNCTIONAL DESCRIPTION:
354 0349 1
355 0350 1 This routine performs common list function for both singular and
356 0351 1 plural entity operations.
357 0352 1
358 0353 1 FORMAL PARAMETERS:
359 0354 1
360 0355 1 ENTITY Entity type code.
361 0356 1 RECDSC Address of permanent data base record descriptor.
362 0357 1 TABDSC Address of parameter list table descriptor.
363 0358 1
364 0359 1 IMPLICIT INPUTS:
365 0360 1
366 0361 1 NONE
367 0362 1
368 0363 1 IMPLICIT OUTPUTS:
369 0364 1
370 0365 1 NONE
371 0366 1
372 0367 1 ROUTINE VALUE:
373 0368 1 COMPLETION CODES:
374 0369 1
375 0370 1 Always returns success.
376 0371 1
377 0372 1 SIDE EFFECTS:
378 0373 1
379 0374 1 NONE
380 0375 1
381 0376 1 --
382 0377 1
383 0378 2 BEGIN
384 0379 2
385 0380 2 LOCAL
386 0381 2 MSG_SIZE; ! Message size
387 0382 2
388 0383 2
389 0384 2 Get the record owner.
390 0385 2
391 0386 2 NML$AB_MSGBLOCK [MSB$B_CODE] = NMA$C_STS_SUC;
392 0387 2 NML$Q_ENTITYDSC [DSC$W_LENGTH] = NML$K_ENTBUFLN; ! Initialize entity descriptor
393 0388 2 NML$Q_ENTITYDSC [DSC$A_POINTER] = NML$T_ENTITYBUF;
394 0389 2
395 0390 2 IF NML$GETRECOWNER (.RECDSC,
396 0391 2 .ENTITY,
397 0392 2 NML$Q_ENTITYDSC,
398 0393 2 NML$Q_ENTITYDSC [DSC$W_LENGTH])
399 0394 2 THEN
400 0395 2 BEGIN
401 0396 2 NML$AB_MSGBLOCK [MSB$L_FLAGS] = MSB$M_ENTD_FLD; ! Set entity descriptor flag
402 0397 2 NML$AB_MSGBLOCK [MSB$A_ENTITY] = NML$Q_ENTITYDSC; ! Add entity descriptor pointer
403 0398 2 END;
404 0399 2
405 0400 2 NML$BLD_REPLY (NML$AB_MSGBLOCK, MSG_SIZE); ! Build message
```


NML\$LIST
VO4-000

NML LIST permanent parameter module
NML_LISTENTITY List known entity parameters

L 7
16-Sep-1984 00:18:35
14-Sep-1984 12:50:10

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[NML.SRC]NMLLIST.B32;1
Page 13
(6)

```
: 406      0401 2
: 407      0402 2
: 408      0403 2
: 409      0404 2
: 410      0405 2
: 411      0406 2
: 412      0407 2
: 413      0408 1

NML$READPARLIST (NML$GQ_SNDBFDSC, MSG_SIZE, .TABDSC, .RECDSC);
NML$SEND (NML$AB_SNDBUFFER, .MSG_SIZE); ! Send message
RETURN NML$_STS_SUC
END;                                     ! End of NML_LISTENTITY
```

```
000C 00000 NML_LISTENTITY:
      53 00000000G 00 9E 00002 .WORD Save R2,R3 : 0345
      52 00000000' 00 9E 00009 MOVAB NML$AB_MSGBLOCK, R3
      5E 00000000' 04 C2 00010 MOVAB NML$Q_ENTITYDSC, R2
04 A3 00000000' 01 90 00013 SUBL2 #4, SP
      62 40 8F 9B 00017 MOVZBW #1, NML$AB_MSGBLOCK+4 : 0386
04 A2 C0 A2 9E 0001B MOVAB #64, NML$Q_ENTITYDSC : 0387
      52 DD 00020 PUSHL R2 NML$T_ENTITYBUF, NML$Q_ENTITYDSC+4 : 0388
      52 DD 00022 PUSHL R2 : 0393
      04 AC DD 00024 PUSHL ENTITY : 0390
      08 AC DD 00027 PUSHL RECDSC : 0393
00000000G 00 04 FB 0002A CALLS #4, NML$GETRECOWNER
      07 50 E9 00031 BLBC R0, 1$
      63 10 D0 00034 MOVL #16, NML$AB_MSGBLOCK : 0396
14 A3 62 9E 00037 MOVAB NML$Q_ENTITYDSC, NML$AB_MSGBLOCK+20 : 0397
00000000G 00 4008 8F BB 0003B 1$: PUSH R3, SP : 0400
      08 02 FB 0003F CALLS #2, NML$BLD_REPLY
      0C AC DD 00046 PUSHL RECDSC : 0402
      08 AE 9F 0004C PUSHL TABDSC
00000000G 00 00000000G 00 9F 0004F PUSHAB MSG_SIZE
      04 FB 00055 CALLS #4, NML$READPARLIST : 0404
      6E DD 0005C PUSHL MSG_SIZE
00000000G 00 00000000G 00 9F 0005E PUSHAB NML$AB_SNDBUFFER
      02 FB 00064 CALLS #2, NML$SEND : 0406
      50 01 D0 0006B MOVL #1, R0 : 0408
      04 0006E RET
```

; Routine Size: 111 bytes, Routine Base: \$CODE\$ + 0162

```

: 415 0409 1 %SBTTL 'NML$LIST TYPE_NODES List known node parameters'
: 416 0410 1 GLOBAL ROUTINE NML$LIST_TYPE_NODES (ENTITY, INF, DUM1, DUM2) : NOVALUE =
: 417 0411 1
: 418 0412 1 !++
: 419 0413 1 FUNCTIONAL DESCRIPTION:
: 420 0414 1 This routine lists permanent data base information for known remote
: 421 0415 1 nodes or loopnodes.
: 422 0416 1
: 423 0417 1 FORMAL PARAMETERS:
: 424 0418 1
: 425 0419 1 ENTITY Entity type code (always NML$C_NODE).
: 426 0420 1 INF Information type code.
: 427 0421 1 DUM1 Not used.
: 428 0422 1 DUM2 Not used.
: 429 0423 1
: 430 0424 1 --
: 431 0425 1
: 432 0426 2 BEGIN
: 433 0427 2
: 434 0428 2 LOCAL
: 435 0429 2 recdsc : DESCRIPTOR, ! Record descriptor
: 436 0430 2 tabdsc : REF DESCRIPTOR, ! Table descriptor
: 437 0431 2 rewind_flag;
: 438 0432 2
: 439 0433 2
: 440 0434 2 Get entity information.
: 441 0435 2
: 442 0436 2 nml$gettext (.entity, .inf, tabdsc); ! Get information table
: 443 0437 2
: 444 0438 2 Read all records that have a node type key (remote or loopnode) that
: 445 0439 2 matches the entity type parameter.
: 446 0440 2
: 447 0441 2 rewind_flag = true;
: 448 0442 2 WHILE nml$read_known_node_rec (.entity,
: 449 0443 2 nml$gq_recbfdsc,
: 450 0444 2 recdsc,
: 451 0445 2 .rewind_flag) DO
: 452 0446 3 BEGIN
: 453 0447 3 rewind_flag = false;
: 454 0448 3 nml$listentity (.entity, recdsc, .tabdsc);
: 455 0449 2 END;
: 456 0450 1 END; ! End of nml$list_type_nodes
```

			0004 0000	.ENTRY	NML\$LIST_TYPE_NODES, Save R2	: 0410
	5E		0C C2 00002	SUBL2	#12, SP	: 0436
			5E DD 00005	PUSHL	SP	:
	7E	04	AC 7D 00007	MOVQ	ENTITY, -(SP)	:
00000000V	00		03 FB 0000B	CALLS	#3, NML\$GETTABLE	:
	52		01 D0 00012	MOVL	#1, REWIND_FLAG	: 0441
			52 DD 00015	PUSHL	REWIND_FLAG	: 0445
		08	AE 9F 00017	PUSHAB	RECDSC	: 0442
		00000000G	00 9F 0001A	PUSHAB	NML\$GQ_RECBFDSC	:
		04	AC DD 00020	PUSHL	ENTITY	:

NML\$LIST
V04-000

NML LIST permanent parameter module
NML\$LIST_TYPE_NODES List known node

N 7
16-Sep-1984 00:18:35
14-Sep-1984 12:50:10

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

Page 15
(7)

00000000G 00
11

04 FB 00023
50 E9 0002A
52 D4 0002D
6E DD 0002F
08 AE 9F 00031
04 AC DD 00034
03 FB 00037
D7 11 0003C
04 0003E 2\$:

CALLS #4, NML\$READ_KNOWN_NODE_REC
BLBC R0, 2\$
CLRL REWIND_FLAG
PUSHL TABDSC
PUSHAB RECDSC
PUSHL ENTITY
CALLS #3, NML_LISTENTITY
BRB 1\$
RET

:
:
: 0447
: 0448
:
:
: 0442
: 0450

; Routine Size: 63 bytes, Routine Base: \$CODE\$ + 01D1

```

458 0451 1 %SBTTL 'NML$GETTABLE Get table descriptor'
459 0452 1 GLOBAL ROUTINE NML$GETTABLE (ENTITY, INF, TABDSC) : NOVALUE =
460 0453 1
461 0454 1 !++
462 0455 1 FUNCTIONAL DESCRIPTION:
463 0456 1
464 0457 1 This routine selects the parameter table for processing permanent
465 0458 1 data base information based on the entity type and the information
466 0459 1 type.
467 0460 1
468 0461 1 FORMAL PARAMETERS:
469 0462 1
470 0463 1 ENTITY Entity id code.
471 0464 1 INF Information type code.
472 0465 1 TABDSC Address of table descriptor.
473 0466 1
474 0467 1 IMPLICIT INPUTS:
475 0468 1
476 0469 1 NONE
477 0470 1
478 0471 1 IMPLICIT OUTPUTS:
479 0472 1
480 0473 1 NONE
481 0474 1
482 0475 1 ROUTINE VALUE:
483 0476 1 COMPLETION CODES:
484 0477 1
485 0478 1 NONE
486 0479 1
487 0480 1 SIDE EFFECTS:
488 0481 1
489 0482 1 NONE
490 0483 1
491 0484 1 --
492 0485 1
493 0486 2 BEGIN
494 0487 2
495 0488 2 LOCAL
496 0489 2 ENTTAB : REF VECTOR;
497 0490 2
498 0491 2
499 0492 2 Get the address of this entity's vector of information tables.
500 0493 2
501 0494 2 ENTTAB = .NML$AL_PERMINFTAB [.ENTITY];
502 0495 2
503 0496 2 Select the table according to the information type.
504 0497 2
505 0498 2 .TABDSC = .ENTTAB [.INF];
506 0499 2
507 0500 2 If the table descriptor address is equal to zero then the requested
508 0501 2 information type is not supported for this entity. Signal an error
509 0502 2 indicating invalid function or option.
510 0503 2
511 0504 2 IF ..TABDSC EQLA 0
512 0505 2 THEN
513 0506 2 NML$ERROR_1 (NMA$C_STS_FUN);
514 0507 2
```


NML\$LIST
V04-000

NML LIST permanent parameter module
NML\$GETTABLE Get table descriptor

C 8
16-Sep-1984 00:18:35
14-Sep-1984 12:50:10

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[NML.SRC]NMLLIST.B32;1 Page 17
(8)

: 515
: 516
: 517
0508 2 RETURN NML\$_STS_SUC
0509 2
0510 1 END;

! End of NML\$GETTABLE

				0000	00000		.ENTRY	NML\$GETTABLE, Save nothing	:	0452
	50		04	AC	DO	00002	MOVL	ENTITY, R0	:	0494
	51	00000000G	0040	DO	DO	00006	MOVL	NML\$AL_PERMINFTAB[R0], ENTTAB	:	
	50		08	AC	DO	0000E	MOVL	INF, R0	:	0498
0C	BC		6140	DO	DO	00012	MOVL	(ENTTAB)[R0], @TABDSC	:	
			0A	12		00017	BNEQ	1\$	:	0504
	7E		01	CE		00019	MNEGL	#1, -(SP)	:	0506
00000000G	00		01	FB		0001C	CALLS	#1, NML\$ERROR_1	:	
			04	00023	1\$:		RET		:	0510

; Routine Size: 36 bytes, Routine Base: \$CODE\$ + 0210

; 518 0511 1

NML\$LIST
V04-000

NML LIST permanent parameter module
NML\$GETTABLE Get table descriptor

D 8
16-Sep-1984 00:18:35
14-Sep-1984 12:50:10
! End of module

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

Page 18
(9)

: 520
: 521
: 522

0512 1 END
0513 1
0514 0 ELUDOM

</

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
\$255\$DUA28:[NML.OBJ]NMLLIB.L32;1	341	32	9	27	00:00.1
\$255\$DUA28:[SHRLIB]NMALIBRY.L32;1	887	5	0	47	00:00.2
\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	2	0	581	00:02.2

COMMAND QUALIFIERS

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

Size: 564 code + 76 data bytes

Run Time: 00:12.7

Elapsed Time: 00:39.8

Lines/CPU Min: 2430

Lexemes/CPU-Min: 9243

Memory Used: 100 pages

Compilation Complete

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

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

