

NN	NN	TTTTTTTTTT	000000	RRRRRRRR	EEEEEEEEEE	NN	NN	AAAAAA	MM	MM	EEEEEEEEEE	
NN	NN	TTTTTTTTTT	000000	RRRRRRRR	EEEEEEEEEE	NN	NN	AAAAAA	MM	MM	EEEEEEEEEE	
NN	NN	TT	00	RR	EE	NN	NN	AA	MMM	MMM	EE	
NN	NN	TT	00	RR	EE	NN	NN	AA	MMM	MMM	EE	
NNNN	NN	TT	00	RR	EE	NNNN	NN	AA	MM	MM	EE	
NNNN	NN	TT	00	RR	EE	NNNN	NN	AA	MM	MM	EE	
NN	NN	TT	00	RRRRRRRR	EEEEEEEEEE	NN	NN	AA	MM	MM	EEEEEEEEEE	
NN	NN	TT	00	RRRRRRRR	EEEEEEEEEE	NN	NN	AA	MM	MM	EEEEEEEEEE	
NN	NN	TT	00	RR	EE	NN	NNNN	AAAAAAAAAA	MM	MM	EE	
NN	NN	TT	00	RR	EE	NN	NNNN	AAAAAAAAAA	MM	MM	EE	
NN	NN	TT	00	RR	EE	NN	NN	AA	MM	MM	EE	
NN	NN	TT	00	RR	EE	NN	NN	AA	MM	MM	EE	
NN	NN	TT	000000	RR	EEEEEEEEEE	NN	NN	AA	MM	MM	EEEEEEEEEE	
NN	NN	TT	000000	RR	EEEEEEEEEE	NN	NN	AA	MM	MM	EEEEEEEEEE	

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
LL	II		SS
LLLLLLLLLL	IIIIII	SSSSSSSS	
LLLLLLLLLL	IIIIII	SSSSSSSS	

NTORENAME
Table of contents

NETWORK RENAME

E 5

16-SEP-1984 00:06:13 VAX/VMS Macro V04-00

Page 0

(2) 60
(3) 91

DECLARATIONS
NT\$RENAME - RENAME FILE


```
0000 1          $BEGIN NTORENAME,000,NF$NETWORK,<NETWORK RENAME>
0000 2
0000 3
0000 4
0000 5 :*****
0000 6 :*
0000 7 :*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 :*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 :*  ALL RIGHTS RESERVED.
0000 10 :*
0000 11 :*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 :*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 :*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 :*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 :*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 :*  TRANSFERRED.
0000 17 :*
0000 18 :*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 :*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 :*  CORPORATION.
0000 21 :*
0000 22 :*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 :*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 :*
0000 25 :*
0000 26 :*****
0000 27 :
0000 28 :
0000 29 :++
0000 30 : Facility: RMS
0000 31 :
0000 32 : Abstract:
0000 33 :
0000 34 :     This module communicates with the File Access Listener (FAL) at the
0000 35 :     remote node to rename the specified file.
0000 36 :
0000 37 : Environment: VAX/VMS, executive mode
0000 38 :
0000 39 : Author: James A. Krycka,      Creation Date: 01-JUN-1981
0000 40 :
0000 41 : Modified By:
0000 42 :
0000 43 :     V03-005 JAK0140          J A Krycka      09-APR-1984
0000 44 :     Properly restore registers on exit from routine.
0000 45 :
0000 46 :     V03-004 JAK0138          J A Krycka      28-MAR-1984
0000 47 :     Call modified NT$EXCH_CNF routine with a parameter.
0000 48 :
0000 49 :     V03-003 KRM0095          Karl Malik      01-Apr-1983
0000 50 :     Modify to support DAP V7.0 flavor of rename message exchange.
0000 51 :
0000 52 :     V03-002 KRM0080          Karl Malik      02-Feb-1983
0000 53 :     Fix bug in RECV_CMP.
0000 54 :
0000 55 :     V03-001 KRM0076          Karl Malik      23-Nov-1982
0000 56 :     Support network rename operation based on DAP V6.0 rules.
0000 57 :
```

NTORENAME
V04-000

NETWORK RENAME

G 5

16-SEP-1984 00:06:13
5-SEP-1984 16:21:05

VAX/VMS Macro V04-00
[RMS.SRC]NTORENAME.MAR;1

Page 2
(1)

0000 58 ;--

NT
VO


```

0000 60      .SBTTL  DECLARATIONS
0000 61
0000 62 :
0000 63 : Include Files:
0000 64 :
0000 65
0000 66      $DAPPLGDEF      ; Define DAP prologue symbols
0000 67      $DAPHDRDEF     ; Define DAP message header
0000 68      $DAPCNFDEF     ; Define DAP Configuration message
0000 69      $DAPACCDEF     ; Define DAP access message symbols
0000 70      $DAPNAMDEF     ; Define DAP Name message symbols
0000 71      $FABDEF        ; Define File Access Block symbols
0000 72      $IFBDEF        ; Define Internal Fab block symbols
0000 73      $FWADEF        ; Define File Work Area symbols
0000 74      $NWADEF        ; Define Network Work Area symbols
0000 75 :
0000 76 : Macros:
0000 77 :
0000 78      None
0000 79 :
0000 80 : Equated Symbols:
0000 81 :
0000 82
0000 83      ASSUME  DAP$Q_DCODE_FLG EQ 0
0000 84
0000 85 :
0000 86 : Own Storage:
0000 87 :
0000 88      None
0000 89 :

```

```
0000 91 .SBTTL NT$RENAME - RENAME FILE
0000 92
0000 93 :++
0000 94 : NT$RENAME - engages in a DAP dialogue with the remote FAL to rename the
0000 95 : specified file.
0000 96 :
0000 97 : Calling Sequence:
0000 98 :
0000 99 : BSBW NT$RENAME
0000 100 :
0000 101 : Input Parameters:
0000 102 :
0000 103 : R8 Old FAB address
0000 104 : R9 IFAB address
0000 105 : R10 New FWA address
0000 106 : R11 Impure Area address
0000 107 : 4(SP) Old FWA address
0000 108 : 8(SP) New FAB address
0000 109 : 12(SP) Old FAB address
0000 110 :
0000 111 : Implicit Inputs:
0000 112 :
0000 113 : FWA$Q_NODE1
0000 114 :
0000 115 : Output Parameters:
0000 116 :
0000 117 : R0-R7 Destroyed
0000 118 :
0000 119 : Implicit Outputs:
0000 120 :
0000 121 : None
0000 122 :
0000 123 : Completion Codes:
0000 124 :
0000 125 : Standard RMS completion codes
0000 126 :
0000 127 : Side Effects:
0000 128 :
0000 129 : None
0000 130 :
0000 131 :--
0000 132 :
0000 133 NT$RENAME:: : Entry point
0000 134 $STPT NT$RENAME :
0000 135 MOVL IFB$L_NWA_PTR(R9),R7 : Get address of NWA (and DAP)
0000 136 :
0000 137 :+
0000 138 : Exchange DAP Configuration messages with FAL and determine DAP buffer size.
0000 139 : Also, check to make sure that partner supports both the rename function and
0000 140 : is implemented to the DAP V7.0 specification (as the message exchange rules
0000 141 : for rename were changed from DAP V6.0 to V7.0).
0000 142 :--
0000 143 :
0000 144 MOVL #DAP$K_RENAME,R0 : Denote type of file access
0000 145 BSBW NT$EXCH_CNF : Exchange Configuration messages
0000 146 BLBS R0,10$ : Branch on success
0000 147 BRW EXIT1 : Branch on failure

57 3C A9 D0 0006 135
000A 136
000A 137
000A 138
000A 139
000A 140
000A 141
000A 142
000A 143
50 03 D0 000A 144
FFF0 30 000D 145
03 50 E8 0010 146
00F5 31 0013 147
```



```
06 28 25 E0 0016 148 10$: BBS #DAP$V_RENAME, - ; Branch if partner supports rename
    A7 0018 149 DAP$Q_SYSCAP(R7),30$ ; operation
    FFE2' 30 001B 150 20$: BSBW NT$RMT_ACCFUNC ; Declare RMS$_SUPPORT error
    00EA 31 001E 151 BRW EXIT1 ; Branch and exit with RMS code in R0
F6 67 26 E1 0021 152 30$: BBC #DAP$V_GEQ_V70,(R7),20$ ; Branch if partner does not support
    0025 153 ; DAP V7.0 message exchange rules
    0025 154
    0025 155 ;+
    0025 156 ; Next build a request mask (NWA$W_DISPLAY) that will be used in the Access
    0025 157 ; message to request that optional DAP messages be returned by FAL. Since only
    0025 158 ; one display field can be sent (in the Access message), use the old NAM block
    0025 159 ; to determine this.
    0025 160 ; -
    0025 161
    0025 162 BUILD_MASK: ; Build NWA$W_DISPLAY
    52 D4 0025 163 CLRL R2 ; Initialize request mask
    FFD6' 30 0027 164 BSBW NT$SCAN_NAMBLK ; Scan user Name Block and check FAL's
    002A 165 ; capabilities to update request mask
    03 50 E8 002A 166 BLBS R0,10$ ; Branch on success
    00DB 31 002D 167 BRW EXIT1 ; Branch on failure
00D0 C7 52 B0 0030 168 10$: MOVW R2,NWA$W_DISPLAY(R7) ; Save request mask
    0035 169
    0035 170 ;+
    0035 171 ; Just check the new NAM block for validity (no request mask is generated).
    0035 172 ; -
    0035 173
    57 08 AE D0 0035 174 MOVL 8(SP),R7 ; Get new FAB address
    57 28 A7 D0 0039 175 MOVL FAB$L_NAM(R7),R7 ; Get new NAM2 address
    FFC0' 30 003D 176 BSBW RM$CHRNAM ; Check NAM2 block for validity
    03 50 E8 0040 177 BLBS R0,20$ ; Branch on success
    00C5 31 0043 178 BRW EXIT1 ; Branch on failure
    57 3C A9 D0 0046 179 20$: MOVL IFB$L_NWA_PTR(R9),R7 ; Restore NWA/DAP address
    004A 180
    004A 181 ;+
    004A 182 ; Build and send DAP Access message to partner.
    004A 183 ; -
    004A 184
    004A 185 SEND_ACC: ; (required message)
    56 5A DD 004A 186 PUSHL R10 ; Save the new FWA address
    0C AE D0 004C 187 MOVL 12(SP),R6 ; Get new FAB address
    5A 08 AE D0 0050 188 MOVL 8(SP),R10 ; Get old FWA address
    50 03 D0 0054 189 MOVL #DAP$K_ACC_MSG,R0 ; Get message type value
    FFA6' 30 0057 190 BSBW NT$BUIED_HEAD ; Construct message header
    85 03 90 005A 191 MOVW #DAP$K_RENAME,(R5)+ ; Store ACCFUNC field
    85 01 90 005D 192 MOVW #DAP$M_NONFATAL,(R5)+ ; Store ACCOPT field
    FF9D' 30 0060 193 BSBW NT$GET_FILESPEC ; Store filespec as a counted string
    51 00D0 C7 3C 0063 194 MOVZWL NWA$W_DISPLAY(R7),R1 ; Get request mask
    05 13 0068 195 BEQL 10$ ; Branch if nothing set
    85 B4 006A 196 CLRW (R5)+ ; Store dummy FAC and SHR fields
    FF91' 30 006C 197 BSBW NT$CVT_BN4_EXT ; Store DISPLAY as an extensible field
    FF8E' 30 006F 198 10$: BSBW NT$BUIED_TAIL ; Finish building message
    FF8B' 30 0072 199 BSBW NT$TRANSMIT ; Send Access message to FAL
    03 50 E8 0075 200 BLBS R0,SEND_NEWNAM ; Branch on success
    0089 31 0078 201 BRW EXIT ; Branch on failure
    007B 202
    007B 203 ;+
    007B 204 ; Build and send DAP Name message to partner (containing the NEW filespec).
```



```
007B 205 :-
007B 206
007B 207 SEND_NEWNAM: ; (required message)
58 5A 6E D0 007B 208      MOVL      (SP),R10 ; Get address of new filespec FWA
   OC AE D0 007E 209      MOVL      12(SP),R8 ; Get address of new filespec FAB
   50 0F D0 0082 210      $SETBIT #NWA$V_LAST MSG,(R7) ; Declare this last message to block
   FF74' 30 0086 211      MOVL      #DAP$K_NAM MSG,R0 ; Get message type value
85 01 90 0089 212      BSBW      NT$BUICD HEAD ; Construct message header
   FF6E' 30 008C 213      MOVW      #DAP$M_FILSPEC,(R5)+ ; Store NAMETYPE field
   FF6B' 30 008F 214      BSBW      NT$GET_FILESPEC ; Store filespec as a counted string
   FF68' 30 0092 215      BSBW      NT$BUICD TAIL ; Finish building message
   FF68' 30 0095 216      BSBW      NT$TRANSMIT ; Send Name message to FAL
69 50 E9 0098 217      BLBC      R0,EXIT ; Branch on failure
009B 218
009B 219 ;+
009B 220 ; Receive DAP Name message (containing the OLD filespec) from partner.
009B 221 :-
009B 222
009B 223 RECV_OLDNAM: ; (required message)
0000 C7 B5 009B 224      TSTW      NWA$W_DISPLAY(R7) ; Branch if Name message was not
   5B 13 009F 225      BEQL      RECV_CMP ; requested
00A1 226      $SETBIT #DAP$K_NAM_MSG,DAP$L_MSG_MASK(R7)
00A6 227 ; Expect response of Name message
58 10 AE D0 00A6 228      MOVL      16(SP),R8 ; Get old FAB address
5A 08 AE D0 00AA 229      MOVL      8(SP),R10 ; Get old FWA address
   FF4F' 30 00AE 230      BSBW      NT$RECEIVE ; Get reply from FAL
   50 50 E9 00B1 231      BLBC      R0,EXIT ; Branch on failure
   FF49' 30 00B4 232      BSBW      NT$DECODE NAM ; Process resultant name string
   FF46' 30 00B7 233      BSBW      RMS$FILLNAM ; Fill in NAM block (if any)
47 50 E9 00BA 234      BLBC      R0,EXIT ; Branch on failure
00BD 235
00BD 236 ;+
00BD 237 ; Receive DAP Acknowledge message from partner.
00BD 238 :-
00BD 239
00BD 240 RECV_ACK1: ; (required message)
57 3C A9 D0 00BD 241      MOVL      IFB$L_NWA_PTR(R9),R7 ; Restore NWA/DAP address
   00C1 242      $SETBIT #DAP$R_ACR_MSG,DAP$L_MSG_MASK(R7)
   00C6 243 ; Expect response of Acknowledge message
   FF37' 30 00C6 244      BSBW      NT$RECEIVE ; Get reply from FAL
   38 50 E9 00C9 245      BLBC      R0,EXIT ; Branch on failure
00CC 246
00CC 247 ;+
00CC 248 ; Receive DAP Name message (containing the NEW filespec) from partner.
00CC 249 :-
00CC 250
00CC 251 RECV_NEWNAM: ; (required message)
0000 C7 B5 00CC 252      TSTW      NWA$W_DISPLAY(R7) ; Branch if Name message was not
   2A 13 00D0 253      BEQL      RECV_CMP ; requested
00D2 254      $SETBIT #DAP$K_NAM_MSG,DAP$L_MSG_MASK(R7)
00D7 255 ; Expect response of Name message
58 0C AE D0 00D7 256      MOVL      12(SP),R8 ; Get new FAB address
5A 6E D0 00DB 257      MOVL      (SP),R10 ; Get new FWA address
   FF1F' 30 00DE 258      BSBW      NT$RECEIVE ; Get reply from FAL
   20 50 E9 00E1 259      BLBC      R0,EXIT ; Branch on failure
   FF19' 30 00E4 260      BSBW      NT$DECODE NAM ; Process resultant name string
   FF16' 30 00E7 261      BSBW      RMS$FILLNAM ; Fill in NAM block (if any)
```

```
17 50 E9 00EA 262 BLBC R0,EXIT ; Branch on failure
      00ED 263
      00ED 264 ;+
      00ED 265 ; Receive DAP Acknowledge from partner.
      00ED 266 ;+
      00ED 267 ;+
      00ED 268 RECV_ACK2: ; (required message)
57 3C A9 D0 00ED 269 MOVL IFB$NWA_PTR(R9),R7 ; Restore NWA/DAP address
      00F1 270 $SETBIT #DAP$R_ACR_MSG,DAP$NMSG_MASK(R7)
      00F6 271 ; Expect response of Acknowledge message
      FF07' 30 00F6 272 BSBW NT$RECEIVE ; Get reply from FAL
08 50 E9 00F9 273 BLBC R0,EXIT ; Branch on failure
      00FC 274
      00FC 275 ;+
      00FC 276 ; Receive DAP Access Complete message from partner.
      00FC 277 ;+
      00FC 278 ;+
      00FC 279 RECV_CMP: ; (required message)
      00FC 280 $SETBIT #DAP$K_CMP_MSG,DAP$NMSG_MASK(R7)
      0101 281 ; Expect response of Access Complete msg
      FEFC' 30 0101 282 BSBW NT$RECEIVE ; Get reply from FAL
      0104 283
58 10 AE D0 0104 284 EXIT: MOVL 16(SP),R8 ; Restore old FAB (as on entry)
      5A 8ED0 0108 285 POPL R10 ; Restore new FWA (as on entry)
      05 010B 286 EXIT1: RSB ; Exit with RMS code in R0
      010C 287
      010C 288 .END ; End of module
```


NTORENAME
Symbol table

NETWORK RENAME

M 5

16-SEP-1984 00:06:13 VAX/VMS Macro V04-00
5-SEP-1984 16:21:05 [RMS.SRC]NTORENAME.MAR;1

Page 8
(3)

\$\$PSECT EP	= 00000000	
\$\$RMSTEST	= 0000001A	
\$\$RMS_PBUGCHK	= 00000010	
\$\$RMS_TBUGCHK	= 00000008	
\$\$RMS_UMODE	= 00000004	
BUILD_MASK	00000025	R 01
DAP\$B_ACCFUNC	00000040	
DAP\$B_ACCOPT	00000041	
DAP\$B_BITCNT	00000035	
DAP\$B_DCODE_FID	00000019	
DAP\$B_DCODE_MAC	0000001B	
DAP\$B_DCODE_MSG	0000001A	
DAP\$B_DECVER	00000047	
DAP\$B_ECONUM	00000045	
DAP\$B_FAC	00000042	
DAP\$B_FILESYS	00000043	
DAP\$B_FLAGS	00000031	
DAP\$B_LEN256	00000034	
DAP\$B_LENGTH	00000033	
DAP\$B_NAME TYPE	00000040	
DAP\$B_OSType	00000042	
DAP\$B_SHR	00000043	
DAP\$B_STREAMID	00000032	
DAP\$B_Type	00000030	
DAP\$B_USRNUM	00000046	
DAP\$B_USRVER	00000048	
DAP\$B_VERNUM	00000044	
DAP\$B_X_FIELD	00000024	
DAP\$C_BLN	000000C0	
DAP\$K_ACC_MSG	= 00000003	
DAP\$K_ACK_MSG	= 00000006	
DAP\$K_BLN	000000C0	
DAP\$K_CMP_MSG	= 00000007	
DAP\$K_NAM_MSG	= 0000000F	
DAP\$K_RENAME	= 00000003	
DAP\$K_CMW	00000030	
DAP\$K_CRC_RSLT	00000020	
DAP\$K_DCODE_STS	00000018	
DAP\$K_MSG_MASK	0000001C	
DAP\$K_SSPWA	00000080	
DAP\$K_TEMP	00000090	
DAP\$M_BITCNT	= 00000008	
DAP\$M_DFTSPEC	= 00000010	
DAP\$M_DSP_3NAM	= 00000200	
DAP\$M_FILSPEC	= 00000001	
DAP\$M_GET	= 00000002	
DAP\$M_GO_NOGO	= 00000010	
DAP\$M_MSE	= 00000010	
DAP\$M_NONFATAL	= 00000001	
DAP\$M_SEGMENT	= 00000040	
DAP\$M_TMP1\$	= 00000020	
DAP\$M_TMP2\$	= 000000C0	
DAP\$Q_DCODE_FLG	00000000	
DAP\$Q_FILESPEC	00000044	
DAP\$Q_MSG_BUF1	00000008	
DAP\$Q_MSG_BUF2	00000010	
DAP\$Q_NAMESPEC	00000044	

DAP\$Q_PASSWORD	00000050	
DAP\$Q_SYSCAP	00000028	
DAP\$Q_SYSPEC	00000038	
DAP\$V_GEQ_V70	= 00000026	
DAP\$V_RENAME	= 00000025	
DAP\$W_BUFSIZ	00000040	
DAP\$W_DISPLAY1	0000004C	
DAP\$W_PARTNER	00000006	
DAP\$W_VERSION	00000004	
EXIT	00000104	R 01
EXIT1	0000010B	R 01
FAB\$K_NAM	= 00000028	
IFB\$K_NWA_PTR	= 0000003C	
NT\$BUILD_HEAD	*****	X 01
NT\$BUILD_TAIL	*****	X 01
NT\$CVT_BN4_EXT	*****	X 01
NT\$DECODE_NAM	*****	X 01
NT\$EXCH_CNF	*****	X 01
NT\$GET_FILESPEC	*****	X 01
NT\$RECEIVE	*****	X 01
NT\$RENAME	00000000	RG 01
NT\$RMT_ACCFUNC	*****	X 01
NT\$SCAN_NAMBLK	*****	X 01
NT\$TRANSMIT	*****	X 01
NWASB_ALLXABCNT	0000011C	
NWASB_DAP_RAC	000000C9	
NWASB_FILESYS	000000C5	
NWASB_KEYXABCNT	0000011D	
NWASB_NETSTRSIZ	0000016F	
NWASB_NODBUFSIZ	00000168	
NWASB_ORG	000000C6	
NWASB_OSType	000000C4	
NWASB_RFM	000000C7	
NWASB_RMS_RAC	000000C8	
NWASB_BLN	00000800	
NWASB_BLN	00000800	
NWASL_ALLXABADR	00000100	
NWASL_DATXABADR	00000104	
NWASL_DEV	000000C0	
NWASL_FHCXABADR	00000108	
NWASL_KEYXABADR	0000010C	
NWASL_MSG_MASK	000000D4	
NWASL_PROXABADR	00000110	
NWASL_RDTXABADR	00000114	
NWASL_SAVE_FLGS	00000128	
NWASL_SUMXABADR	00000118	
NWASL_THREAD	000000FC	
NWASL_XLTATTR	00000238	
NWASL_XLTBUFFLG	0000022C	
NWASL_XLTCNT	00000228	
NWASL_XLTMAXINDX	00000234	
NWASL_XLTSIZ	00000230	
NWASQ_ACS	00000244	
NWASQ_BIGBUF	00000170	
NWASQ_BLD	000000F0	
NWASQ_FLG	00000000	
NWASQ_INODE	0000025C	

NT
VO

NTORENAME
Symbol table

NETWORK RENAME

N 5

16-SEP-1984 00:06:13
5-SEP-1984 16:21:05

VAX/VMS Macro V04-00
[RMS.SRC]NTORENAME.MAR;1

Page 9
(3)

NWASQ_IOSB	000000D8		
NWASQ_LNODE	00000160		
NWASQ_LOGNAME	0000023C		
NWASQ_NCB	00000264		
NWASQ_RCV	000000E0		
NWASQ_SAVE_DESC	00000120		
NWASQ_XLTBUF1	0000024C		
NWASQ_XLTBUF2	00000254		
NWASQ_XMT	000000E8		
NWAST_ACSBUF	0000026C		
NWAST_AUXBUF	000005E0		
NWAST_DAP	00000000		
NWAST_INODEBUF	000004AC		
NWAST_ITM_ATTR	00000200		
NWAST_ITM_END	00000224		
NWAST_ITM_LST	00000200		
NWAST_ITM_MAXINDX	00000218		
NWAST_ITM_STRING	0000020C		
NWAST_NCBBUF	0000052C		
NWAST_NODEBUF	00000169		
NWAST_RCVBUF	000001A0		
NWAST_SCAN	00000100		
NWAST_TEMP	00000120		
NWAST_XLTBUF1	000002AC		
NWAST_XLTBUF2	000003AC		
NWAST_XMTBUF	000003C0		
NWASV_LAST_MSG	= 00000000		
NWASW_BUILD	000000D2		
NWASW_DAPBUFSIZ	000000CA		
NWASW_DIR_OFF	000000CC		
NWASW_DISPLAY	000000D0		
NWASW_FIL_OFF	000000CE		
NWASW_JNLXABJOP	0000011E		
PIOSA_TRACE	*****	X	01
RECV_ACK1	000000BD	R	01
RECV_ACK2	000000ED	R	01
RECV_CMP	000000FC	R	01
RECV_NEWNAM	000000CC	R	01
RECV_OLDNAM	0000009B	R	01
RMSCHKNAM	*****	X	01
RMSFILLNAM	*****	X	01
SEND_ACC	0000004A	R	01
SEND_NEWNAM	0000007B	R	01
TPT\$C_NTRENAME	*****	X	01

! Psect synopsis !

PSECT name

ABS
NFSNETWORK
\$ABSS

Allocation		PSECT No.	Attributes													
00000000	(0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE			
0000010C	(268.)	01 (1.)	PIC	USR	CON	REL	GBL	NOSHR	EXE	RD	NOWRT	NOVEC	BYTE			
00000800	(2048.)	02 (2.)	NOPIC	USR	CON	ABS	LCL	NOSHR	EXE	RD	WRT	NOVEC	BYTE			

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:02.00
Command processing	106	00:00:00.64	00:00:05.44
Pass 1	323	00:00:10.99	00:00:32.48
Symbol table sort	0	00:00:01.51	00:00:02.70
Pass 2	71	00:00:01.90	00:00:04.56
Symbol table output	19	00:00:00.16	00:00:00.43
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	552	00:00:15.32	00:00:47.63

The working set limit was 1350 pages.
57842 bytes (113 pages) of virtual memory were used to buffer the intermediate code.
There were 60 pages of symbol table space allocated to hold 1096 non-local and 12 local symbols.
288 source lines were read in Pass 1, producing 14 object records in Pass 2.
21 pages of virtual memory were used to define 20 macros.

! Macro library statistics !

Macro library name	Macros defined
_\$255\$DUA28:[RMS.OBJ]RMS.MLB;1	12
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	16

1303 GETS were required to define 16 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:NTORENAME/OBJ=OBJ\$:NTORENAME MSRC\$:NTORENAME/UPDATE=(ENH\$:NTORENAME)+LIB\$:RMS/LIB

0317 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

