

```

RRRRRRRRRRRRRRR      MMM      MMM      SSSSSSSSSSSSSS
RRRRRRRRRRRRRRR      MMM      MMM      SSSSSSSSSSSSSS
RRRRRRRRRRRRRRR      MMM      MMM      SSSSSSSSSSSSSS
RRR      RRR      MMMMMM      MMMMMM      SSS
RRR      RRR      MMMMMM      MMMMMM      SSS
RRR      RRR      MMMMMM      MMMMMM      SSS
RRR      RRR      MMM      MMM      MMM      SSS
RRR      RRR      MMM      MMM      MMM      SSS
RRR      RRR      MMM      MMM      MMM      SSS
RRRRRRRRRRRRRRR      MMM      MMM      SSSSSSSSSS
RRRRRRRRRRRRRRR      MMM      MMM      SSSSSSSSSS
RRRRRRRRRRRRRRR      MMM      MMM      SSSSSSSSSS
RRR      RRR      MMM      MMM      MMM      SSS
RRR      RRR      MMM      MMM      MMM      SSS
RRR      RRR      MMM      MMM      MMM      SSS
RRR      RRR      MMM      MMM      MMM      SSS
RRR      RRR      MMM      MMM      MMM      SSS
RRR      RRR      MMM      MMM      SSSSSSSSSSSSSS
RRR      RRR      MMM      MMM      SSSSSSSSSSSSSS
RRR      RRR      MMM      MMM      SSSSSSSSSSSSSS

```

— 52

Syn

NTS

NTS

NTS

1997

NTS

NTS

NTS

NTS

NTS

NTS
NTSNTS
NTSNTS
NTSNTS
NTSNTS
NTS

NTS

NTS

NTS
NTSNTS
NTSNTS
NTS

NYT

100

NTS

NTS
NTSNT
NT

NI
PI

10

1

1

10

100

```
NN      NN      TTTTTTTTTT  000000  EEEEEEEEEEE RRRRRRRR  AAAAAA  SSSSSSSS  EEEEEEEEEEE
NN      NN      TTTTTTTTTT  000000  EEEEEEEEEEE RRRRRRRR  AAAAAA  SSSSSSSS  EEEEEEEEEEE
NN      NN      TT          00        00  EE          RR      RR  AA      AA  SS          EE
NN      NN      TT          00        00  EE          RR      RR  AA      AA  SS          EE
NNNN     NN      TT          00      0000  EE          RR      RR  AA      AA  SS          EE
NNNN     NN      TT          00      0000  EE          RR      RR  AA      AA  SS          EE
NN      NN      TT          00      00  EEEEEEEEE RRRRRRRR  AA      AA  SSSSSS  EEEEEEEEE
NN      NN      TT          00      00  EEEEEEEEE RRRRRRRR  AA      AA  SSSSSS  EEEEEEEEE
NN      NN      TT          0000      00  EE          RR      RR  AAAAAAAAAA SS          EE
NN      NN      TT          0000      00  EE          RR      RR  AAAAAAAAAA SS          EE
NN      NN      TT          00        00  EE          RR      RR  AA      AA  SS          EE
NN      NN      TT          00        00  EE          RR      RR  AA      AA  SS          EE
NN      NN      TT          000000  EEEEEEEEEEE RRR      RR  AA      AA  SSSSSSSS  EEEEEEEEEEE
NN      NN      TT          000000  EEEEEEEEEEE RR      RR  AA      AA  SSSSSSSS  EEEEEEEEEEE
```

```
LL      IIIIIII  SSSSSSSS
LL      IIIIIII  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  IIIIIII  SSSSSSSS
LLLLLLLLLL  IIIIIII  SSSSSSSS
```


(2) 51
(3) 82

DECLARATIONS
NT\$ERASE - PERFORM NETWORK ERASE FUNCTION

```
0000 1          $BEGIN NTOERASE,000,NF$NETWORK,<NETWORK ERASE FILE>
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 at the remote
0000 35 node to erase (delete) the specified file.
0000 36
0000 37 Environment: VAX/VMS, executive mode
0000 38
0000 39 Author: James A. Krycka,      Creation Date: 09-DEC-1977
0000 40
0000 41 Modified By:
0000 42
0000 43 V03-003 JAK0138      J A Krycka      28-MAR-1984
0000 44 Call modified NT$EXCH_CNF routine with a parameter.
0000 45
0000 46 V03-002 KRM0109      K Malik      12-May-1983
0000 47 Update to support DAP V7.0 message exchange sequence.
0000 48
0000 49 --
```



```

0000 51      .SBTTL  DECLARATIONS
0000 52
0000 53 :
0000 54 : Include Files:
0000 55 :
0000 56
0000 57      $DAPPLGDEF      ; Define DAP prologue symbols
0000 58      $DAPHDRDEF     ; Define DAP message header
0000 59      $DAPCNFDEF     ; Define DAP Configuration message
0000 60      $DAPACCDEF     ; Define DAP Access message
0000 61      $DAPNAMDEF     ; Define DAP Name message
0000 62      $IFBDEF        ; Define IFAB symbols
0000 63      $NWADEF        ; Define Network Work Area symbols
0000 64
0000 65 :
0000 66 : Macros:
0000 67 :
0000 68      None
0000 69 :
0000 70 : Equated Symbols:
0000 71 :
0000 72
0000 73      ASSUME  DAP$Q_DCODE_FLG EQ 0
0000 74      ASSUME  NWA$Q_FLG EQ 0
0000 75
0000 76 :
0000 77 : Own Storage:
0000 78 :
0000 79      None
0000 80 :

```

```

0000 82      .SBTTL  NT$ERASE - PERFORM NETWORK ERASE FUNCTION
0000 83
0000 84      :++
0000 85      : NT$ERASE - engages in a DAP dialogue with the remote FAL to erase (delete)
0000 86      : the specified file.
0000 87
0000 88      : Calling Sequence:
0000 89
0000 90      :     BSBW  NT$ERASE
0000 91
0000 92      : Input Parameters:
0000 93
0000 94      :     R8      FAB address
0000 95      :     R9      IFAB address
0000 96      :     R10     FWA address
0000 97      :     R11     Impure Area address
0000 98
0000 99      : Implicit Inputs:
0000 100
0000 101      :     FAB$$_NAM
0000 102      :     IFB$$_NWA_PTR
0000 103
0000 104      : Output Parameters:
0000 105
0000 106      :     R0      Status code (RMS)
0000 107      :     R1-R7   Destroyed
0000 108      :     AP      Destroyed
0000 109
0000 110     : Implicit Outputs:
0000 111
0000 112     :     Resultant name string
0000 113
0000 114     : Completion Codes:
0000 115
0000 116     :     Standard RMS completion codes
0000 117
0000 118     : Side Effects:
0000 119
0000 120     :     None
0000 121
0000 122     :--
0000 123
0000 124     NT$ERASE::
0000 125     $TSTPT  NTERASE      ; Entry point
0000 126     MOVL   IFB$$_NWA_PTR(R9),R7 ; Get address of NWA (and DAP)
0000 127
0000 128     :+
0000 129     : Exchange DAP Configuration messages with FAL and determine DAP buffer size.
0000 130     :-
0000 131
0000 132     MOVL   #DAP$K_ERASE,R0      ; Denote type of file access
0000 133     BSBW   NT$EXCH_CNF          ; Exchange Configuration messages
0000 134     BLBC   R0,EXIT             ; Branch on failure
0000 135
0000 136     :+
0000 137     : Next build a request mask (NWA$W_DISPLAY) that will be used in the Access
0000 138     : message to request that optional DAP messages be returned by FAL. For $ERASE,

```

57 3C A9 D0

50 04 D0
FFF0 30
60 50 E9


```
0013 139 ; the Name message is the only candidate.
0013 140 :-
0013 141
0013 142 BUILD_MASK: ; Build NWA$W_DISPLAY
0013 143 CLRL R2 ; Initialize request mask
52 D4 0013 143 BSBW NT$SCAN_NAMBLK ; Scan user Name Block and check FAL's
FFEB' 30 0015 144 ; capabilities to update request mask
0018 145
58 50 E9 0018 146 BLBC R0,EXIT ; Branch on failure
00D0 C7 52 B0 001B 147 MOVW R2,NWA$W_DISPLAY(R7) ; Save request mask
0020 148
0020 149 ;+
0020 150 ; Build and send DAP Access message to partner.
0020 151 :-
0020 152
0020 153 SEND_ACC:
0020 154 $SETBIT #NWA$V_LAST_MSG,(R7) ; Declare this last message to block
50 03 D0 0024 155 MOVL #DAP$K_ACC_MSG,R0 ; Get message type value
FFD6' 30 0027 156 BSBW NT$BUICD_HEAD ; Construct message header
85 04 90 002A 157 MOVB #DAP$K_ERASE,(R5)+ ; Store ACCFUNC field
85 01 90 002D 158 MOVB #DAP$M_NONFATAL,(R5)+ ; Store ACCOPT field
FFCD' 30 0030 159 BSBW NT$GET_FILESPEC ; Store FILESPEC as a counted
51 00D0 C7 3C 0033 160 ; ASCII string
05 13 0033 161 MOVZWL NWA$W_DISPLAY(R7),R1 ; Get request mask
85 B4 003A 162 BEQL 10$ ; Branch if nothing set
FFC1' 30 003C 163 CLRW (R5)+ ; Store dummy FAC and SHR fields
FFBE' 30 003F 164 BSBW NT$CVT_BN4_EXT ; Store DISPLAY as an extensible field
FFBB' 30 0042 165 10$: BSBW NT$BUICD_TAIL ; Finish building message
2B 50 E9 0045 166 BSBW NT$TRANSMIT ; Send Access message to FAL
0048 167 BLBC R0,EXIT ; Branch on failure
0048 168
0048 169 ;+
0048 170 ; Receive DAP (resultant) Name message from partner.
0048 171 :-
0048 172
0048 173 RECV_NAM:
0048 174 TSTW NWA$W_DISPLAY(R7) ; Branch if Name message was not
004C 175 BEQL RECV_CMP ; requested
004E 176 $SETBIT #DAP$K_NAM_MSG,DAP$L_MSG_MASK(R7) ; Expect response of Name message
FFAA' 30 0053 177 BSBW NT$RECEIVE ; Get reply from FAL
1A 50 E9 0056 179 BLBC R0,EXIT ; Branch on failure
FFA4' 30 0059 180 BSBW NT$DECODE_NAM ; Process resultant name string
005C 181
005C 182 ;+
005C 183 ; Receive DAP Acknowledge message from partner.
005C 184 :-
005C 185
005C 186 RECV_ACK:
67 26 E1 005C 187 BBC #DAP$V_GEQ_V70,(R7),- ; Acknowledge message required if
0B 005F 188 RECV_CMP ; partner uses DAP since V7.0
0060 189 $SETBIT #DAP$K_ACK_MSG,DAP$L_MSG_MASK(R7) ; Expect response of Acknowledge message
FF98' 30 0065 190 BSBW NT$RECEIVE ; Get reply from FAL
0B 50 E9 0068 191 BLBC R0,EXIT ; Branch on failure
006B 192
006B 193 ;+
006B 194 ; Receive DAP Access Complete message from partner to terminate the reply
006B 195
```


[illegible]

\$\$PSECT EP	=	00000000	
\$\$RMSTEST	=	0000001A	
\$\$RMS_PBUGCHK	=	00000010	
\$\$RMS_TBUGCHK	=	00000008	
\$\$RMS_UMODE	=	00000004	
BUILD_MASK		00000013	R
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_ERASE	=	00000004	
DAP\$K_NAM_MSG	=	0000000F	
DAP\$L_CMWA		00000030	
DAP\$L_CRC_RSLT		00000020	
DAP\$L_DCODE_STS		00000018	
DAP\$L_MSG_MASK		0000001C	
DAP\$L_SSPQA		00000080	
DAP\$L_TEMP		00000090	
DAP\$M_BITCNT	=	00000008	
DAP\$M_DFTSPEC	=	00000010	
DAP\$M_DSP_3NAM	=	00000200	
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	

Variable	Value	Attribute	Mode
DAP\$Q_SYSCAP	00000028		
DAP\$Q_SYSPEC	00000038		
DAP\$V_GEQ_V70	= 00000026		
DAP\$W_BUF\$IZ	00000040		
DAP\$W_DISPLAY1	0000004C		
DAP\$W_PARTNER	00000006		
DAP\$W_VERSION	00000004		
EXIT	00000073	R	01
IFB\$L_NWA_PTR	= 0000003C		
NT\$BUILD_READ	*****	X	0
NT\$BUILD_TAIL	*****	X	01
NT\$CVT_BN4_EXT	*****	X	01
NT\$DECODE_NAM	*****	X	01
NT\$ERASE	00000000	RG	01
NT\$EXCH_CNF	*****	X	01
NT\$GET_FILESPEC	*****	X	01
NT\$RECEIVE	*****	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_NETSTR\$IZ	0000016F		
NWASB_NODBUFSIZ	00000168		
NWASB_ORG	000000C6		
NWASB_OSTYPE	000000C4		
NWASB_RFM	000000C7		
NWASB_RMS_RAC	000000C8		
NWASB_BLN	00000800		
NWASK_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		
NWASQ_IOSB	000000D8		
NWASQ_LNODE	00000160		
NWASQ_LOGNAME	0000023C		
NWASQ_NCB	00000264		
NWASQ_RCV	000000E0		

[illegible]

NTOERASE
Symbol table

NETWORK ERASE FILE

E 12

15-SEP-1984 23:59:56 VAX/VMS Macro V04-00
5-SEP-1984 16:20:46 [RMS.SRC]NTOERASE.MAR;1

Page 7
(3)

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_ACK	0000005C R 01
RECV_CMP	0000006B R 01
RECV_NAM	00000048 R 01
SEND_ACC	00000020 R 01
TPT\$C_NTERASE	***** X 01

+-----+
! Psect synopsis !
+-----+

PSECT name	Allocation	PSECT No.	Attributes														
ABS	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE				
NF\$NETWORK	00000074 (116.)	01 (1.)	PIC	USR	CON	REL	GBL	NOSHR	EXE	RD	NOWRT	NOVEC	BYTE				
\$ABSS	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	32	00:00:00.11	00:00:01.47
Command processing	113	00:00:00.57	00:00:05.58
Pass 1	231	00:00:07.04	00:00:19.43
Symbol table sort	0	00:00:00.84	00:00:01.04
Pass 2	56	00:00:01.20	00:00:03.23
Symbol table output	18	00:00:00.14	00:00:00.37

Psect synopsis output	1	00:00:00.02	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	453	00:00:09.93	00:00:31.53

The working set limit was 1200 pages.

33321 bytes (66 pages) of virtual memory were used to buffer the intermediate code.

There were 40 pages of symbol table space allocated to hold 589 non-local and 5 local symbols.

205 source lines were read in Pass 1, producing 13 object records in Pass 2.

19 pages of virtual memory were used to define 18 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
-----	-----
_\$255\$DUA28:[RMS.OBJ]RMS.MLB;1	10
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	14

794 GETS were required to define 14 macros.

There were no errors, warnings or information messages.

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

0316 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY