

SSSSSSSS	AAAAAA	UU	UU	DDDDDDDD	IIIIII	TTTTTTTTTT	
SSSSSSSS	AAAAAA	UU	UU	DDDDDDDD	IIIIII	TTTTTTTTTT	
SS	AA	AA	UU	UU	DD	DD	
SS	AA	AA	UU	UU	DD	DD	
SS	AA	AA	UU	UU	DD	DD	
SS	AA	AA	UU	UU	DD	DD	
SSSSSS	AA	AA	UU	UU	DD	DD	
SSSSSS	AA	AA	UU	UU	DD	DD	
	AAAAAAAA	UU	UU	DD	DD	DD	
	AAAAAAAA	UU	UU	DD	DD	DD	
	AA	AA	UU	UU	DD	DD	
	AA	AA	UU	UU	DD	DD	
	AA	AA	UU	UU	DD	DD	
SSSSSSSS	AA	AA	UUUUUUUUUU	DDDDDDDD	IIIIII	TTTTTTTTTT
SSSSSSSS	AA	AA	UUUUUUUUUU	DDDDDDDD	IIIIII	TTTTTTTTTT

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 'EDTSSAUDIT - Terminate recovery processing'
2 0002 0 MODULE EDTSSAUDIT ( ! Terminate recovery processing
3 0003 0 IDE.IT = 'V04-000' ! File: SAUDIT.BLI Edit: REM1020
4 0004 0 ) =
5 0005 1 BEGIN
6 0006 1
7 0007 1 *****
8 0008 1 *
9 0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
10 0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
11 0011 1 * ALL RIGHTS RESERVED. *
12 0012 1 *
13 0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
14 0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
15 0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
16 0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
17 0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
18 0018 1 * TRANSFERRED. *
19 0019 1 *
20 0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
21 0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
22 0022 1 * CORPORATION. *
23 0023 1 *
24 0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
25 0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
26 0026 1 *
27 0027 1 *
28 0028 1 *****
29 0029 1
30 0030 1
31 0031 1 **
32 0032 1 FACILITY: EDT -- The DEC Standard Editor
33 0033 1
34 0034 1 ABSTRACT:
35 0035 1
36 0036 1 Terminate recovery processing on the journal file.
37 0037 1
38 0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
39 0039 1
40 0040 1 AUTHOR: Bob Kushlis, CREATION DATE: 6-AUG-1979
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original. DJS 18-FEB-1981. This module was created by
45 0045 1 extracting routine EDT$$TERMIN RCOVMOD from module EDTCTR.
46 0046 1 1-002 - Regularized the headers. JBS 24-Feb-1981
47 0047 1 1-003 - Fixed the file name and module name. JBS 04-Mar-1981
48 0048 1 1-004 - Change EDT$$OPEN APPEND FILE to EDT$$OAPP FILDEF . JBS 02-Apr-1981
49 0049 1 1-005 - Use the journal RAB rather than the command RAB. JBS 16-Aug-1981
50 0050 1 1-006 - Add additional parameters to edt$$cls.fi. STS 15-Jan-1982
51 0051 1 1-007 - Don't close Vax files and then reopen. STS 18-Jan-1982
52 0052 1 1-008 - deallocate dynamic descriptors. STS 10-Feb-1982
53 0053 1 1-009 - Define literals for callable EDT. STS 10-Mar-1982
54 0054 1 1-010 - Change EDT$$G JOU to EDT$$V OPTIONS. JBS 25-Mar-1982
55 0055 1 1-011 - Correct errors in PDP-11 configuration. JBS 26-Mar-1982
56 0056 1 1-012 - The file I/O routines return status. JBS 26-Mar-1982
57 0057 1 1-013 - On the re-open of the PDP-11's journal file, use SEQ to indicate

```



```

: 70 0069 1 %SBTTL 'Declarations'
: 71 0070 1
: 72 0071 1 | TABLE OF CONTENTS:
: 73 0072 1 |
: 74 0073 1
: 75 0074 1 REQUIRE 'EDTSRC:TRAROUNAM';
: 76 0513 1
: 77 0514 1 FORWARD ROUTINE
: 78 0515 1     EDT$$TERMIN_RCOVMOD : NOVALUE;
: 79 0516 1
: 80 0517 1 |
: 81 0518 1 | INCLUDE FILES:
: 82 0519 1 |
: 83 0520 1
: 84 L 0521 1 %IF %BLISS (BLISS32)
: 85 0522 1 %THEN
: 86 0523 1
: 87 0524 1 REQUIRE 'EDTSRC:SYSSYM';
: 88 0554 1
: 89 0555 1 %FI
: 90 0556 1
: 91 0557 1 REQUIRE 'EDTSRC:EDTREQ';
: 92 0692 1
: 93 0693 1 |
: 94 0694 1 | MACROS:
: 95 0695 1 |
: 96 0696 1 |
: 97 0697 1 | +
: 98 0698 1 | Macro for the default name of the journal file. This is defined as a macro
: 99 0699 1 | so we can use %CHARCOUNT to pass the length of the string.
: 100 0700 1 | -
: 101 0701 1 | <BLF/NOFORMAT>
: 102 0702 1 MACRO
: 103 0703 1     JOU_DEFLT = %STRING ('.JOU;') % ; ! File type of .JOU, latest version.
: 104 0704 1
: 105 0705 1 | <BLF/FORMAT>
: 106 0706 1
: 107 0707 1 | EQUATED SYMBOLS:
: 108 0708 1
: 109 0709 1 |     NONE
: 110 0710 1
: 111 0711 1 | OWN STORAGE:
: 112 0712 1
: 113 0713 1 |     NONE
: 114 0714 1
: 115 0715 1 | EXTERNAL REFERENCES:
: 116 0716 1
: 117 0717 1 |     In the routine
```

```

119 0718 1 %SBTTL 'EDTSS$TERMIN_RCOVMOD - Terminate recovery processing'
120 0719 1
121 0720 1 GLOBAL ROUTINE EDTSS$TERMIN_RCOVMOD ! Terminate recovery mode
122 0721 1 : NOVALUE =
123 0722 1
124 0723 1 ++
125 0724 1 FUNCTIONAL DESCRIPTION:
126 0725 1
127 0726 1 Terminate recovery mode. This routine closes the journal file
128 0727 1 being used to recover and re-opens it to record subsequent
129 0728 1 typein. On VMS this is not needed because the journal file
130 0729 1 was opened for input/output.
131 0730 1
132 0731 1 FORMAL PARAMETERS:
133 0732 1
134 0733 1 NONE
135 0734 1
136 0735 1 IMPLICIT INPUTS:
137 0736 1
138 0737 1 EDTSS$V_OPTIONS
139 0738 1 EDTSS$A_JOU_NAM
140 0739 1 EDTSS$G_JOU_NAMLEN
141 0740 1
142 0741 1 IMPLICIT OUTPUTS:
143 0742 1
144 0743 1 EDTSS$G_RCOV_MOD
145 0744 1 EDTSS$G_TIN_OBUFPOS
146 0745 1
147 0746 1 ROUTINE VALUE:
148 0747 1
149 0748 1 NONE
150 0749 1
151 0750 1 SIDE EFFECTS:
152 0751 1
153 0752 1 Closes and re-opens the journal file on PDP-11s, may do nothing on VMS.
154 0753 1
155 0754 1 --
156 0755 1
157 0756 2 BEGIN
158 0757 2
159 0758 2 EXTERNAL ROUTINE
160 0759 2 EDTSS$FIOPN_ERR : NOVALUE,
161 0760 2 EDTSS$FATAL_IDERR : NOVALUE,
162 0761 2 EDTSS$CALLFIO;
163 0762 2
164 L 0763 2 %IF %BLISS (BLISS32)
165 0764 2 %THEN
166 0765 2
167 0766 2 EXTERNAL ROUTINE
168 0767 2 STR$FREE1_DX;
169 0768 2
170 U 0769 2 %ELSE
171 U 0770 2
172 U 0771 2 EXTERNAL LITERAL
173 U 0772 2 EDT$K_OPEN_OUTPUT_SEQ;
174 U 0773 2
175 U 0774 2 MESSAGES ((NOJNLFIL));

```

```

176 0775 2 %FI
177 0776 2
178 0777 2 EXTERNAL
179 0778 2 EDT$$V_OPTIONS : BITVECTOR [32],      ! Options; bit 2 = /NOJOURNAL
180 0779 2 EDT$$A_JOU_NAM,                    ! Name of journal file
181 0780 2 EDT$$G_JOU_NAMLEN,                  ! Length of journal file name
182 0781 2 EDT$$G_TIN_OBUFPOS,                ! Position in journal output buffer
183 0782 2 EDT$$G_RCOV_MOD;                   ! Are we in recovery mode?
184 0783 2
185 0784 2 EXTERNAL LITERAL
186 0785 2 EDTSM_NOJOURNAL,
187 0786 2 EDTSK_CLOSE,
188 0787 2 EDTSK_JOURNAL_FILE;
189 0788 2
190 0789 2 LOCAL
191 0790 2 STATUS,
192 0791 2 FILE_DESC : BLOCK [8, BYTE],
193 0792 2 RHB_DESC : BLOCK [8, BYTE],
194 0793 2 RHB_LEN,
195 0794 2 STS;
196 0795 2 STV;
197 0796 2
198 0797 2 MESSAGES ((JOUFILCLO));
199 0798 2
200 0799 2 !+ This routine is only called if we are in recovery mode.
201 0800 2 !-
202 0801 2 ASSERT (.EDT$$G_RCOV_MOD);
203 0802 2
204 L 0803 2 %IF %BLISS (BLISS32)
205 0804 2 %THEN
206 0805 2 FILE_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
207 0806 2 FILE_DESC [DSC$B_CLASS] = DSC$K_CLASS_D;
208 0807 2 RHB_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
209 0808 2 RHB_DESC [DSC$B_CLASS] = DSC$K_CLASS_D;
210 0809 2 %FI
211 0810 2
212 0811 2 FILE_DESC [DSC$A_POINTER] = 0;
213 0812 2 FILE_DESC [DSC$W_LENGTH] = 0;
214 0813 2 RHB_DESC [DSC$A_POINTER] = 0;
215 0814 2 RHB_DESC [DSC$W_LENGTH] = 0;
216 0815 2 !
217 0816 2 RHB_LEN = %CHARCOUNT (JOU_DEFLT);
218 0817 2 STRING_DESC (RHB_DESC, RHB_LEN, CH$PTR (%PLIT (JOU_DEFLT)));
219 0818 2
220 0819 2 IF ((.EDT$$V_OPTIONS AND EDTSM_NOJOURNAL) NEQ 0)
221 0820 2 THEN
222 0821 2 BEGIN
223 0822 2 STRING_DESC (FILE_DESC, EDT$$G_JOU_NAMLEN, .EDT$$A_JOU_NAM);
224 0823 2 STATUS = EDT$$CALCFIO (EDTSK_CLOSE, EDTSK_JOURNAL_FILE, FILE_DESC, 0);
225 0824 2
226 0825 2 IF ( NOT .STATUS)
227 0826 2 THEN
228 0827 2 BEGIN
229 0828 2 EDT$$FIOPN_ERR (EDTS_JOUFILCLO, FILE_DESC);
230 0829 2 EDT$$FATAL_IOERR ();
231 0830 2 END;
232 0831 2

```

```
233      0832      2      END;
234      0833      2
235      0834      2      %IF %BLISS (BLISS16)
236      0835      2      %THEN
237      0836      2      +
238      0837      2      | If this is a PDP-11 base system, we must close the file and re-open it for append.
239      0838      2      | We use SEQ not to make the journal file sequenced but to indicate that the journal
240      0839      2      | file should be appended to.
241      0840      2      | -
242      0841      2
243      0842      2      IF ((.EDT$$V_OPTIONS AND EDTSM_NOJOURNAL) EQL 0)      ! NOT /NOJOURNAL
244      0843      2      THEN
245      0844      2      BEGIN
246      0845      2      STRING_DESC (FILE_DESC, EDT$$G_JOU_NAMLEN, .EDT$$A_JOU_NAM);
247      0846      2      STATUS = EDT$$CALLFIO (EDT$K_CLOSE, EDT$K_JOURNAL_FILE, 0, 0);
248      0847      2
249      0848      2      IF ( NOT .STATUS)
250      0849      2      THEN
251      0850      2      BEGIN
252      0851      2      EDT$$FIOPN_ERR (EDT$_JOUFILCLO, FILE_DESC);
253      0852      2      EDT$$FATAL_IOERR ();
254      0853      2      END;
255      0854      2
256      0855      2      STATUS = EDT$$CALLFIO (EDT$K_OPEN_OUTPUT_SEQ, EDT$K_JOURNAL_FILE, FILE_DESC, RMB_DESC);
257      0856      2
258      0857      2      IF ( NOT .STATUS)
259      0858      2      THEN
260      0859      2      BEGIN
261      0860      2      EDT$$FIOPN_ERR (EDT$_NOJNLFIL, FILE_DESC);
262      0861      2      EDT$$FATAL_IOERR ();
263      0862      2      END;
264      0863      2
265      0864      2      END;
266      0865      2
267      0866      2      %FI
268      0867      2
269      0868      2      +
270      0869      2      | Recovery mode is over.
271      0870      2      | -
272      0871      2      EDT$$G_RCOV_MOD = 0;
273      0872      2      edt$$g_tin_obufpos = 0;
274      0873      2      +
275      0874      2      | On the VAX we need to deallocate the dynamic descriptors
276      0875      2      | -
277      0876      2
278      0877      2      %IF %BLISS (BLISS32)
279      0878      2      %THEN
280      0879      2      STR$FREE1_DX (FILE_DESC);
281      0880      2      STR$FREE1_DX (RMB_DESC);
282      0881      2      %FI
283      0882      2
284      0883      1      END;
! of routine EDT$$TERMIN_RCOVMOD
```

```
.TITLE EDTSSAUDIT EDTSSAUDIT - Terminate recovery proc
essing
.IDENT \V04-000\
```



```

.PSECT _EDT$CODE,NOWRT, SHR, PIC,2
00 00 00 3B 55 4F 4A 2E 0000 P.AAA: .ASCII \.JOU;\<0><0><0>
;
;EXTRN EDT$$FIOPN_ERR, EDT$$FATAL_IOERR
;EXTRN EDT$$CALLFIO, STR$FREE1_DX
;EXTRN EDT$$V_OPTIONS, EDT$$A_JOU_NAM
;EXTRN EDT$$G_JOU_NAMLEN
;EXTRN EDT$$G-TIN_OBUFPOS
;EXTRN EDT$$G-RCOV_MOD
;EXTRN EDTSM_NOJOURNAL
;EXTRN EDT$K_CLOSE, EDT$K_JOURNAL_FILE
;EXTRN EDT$ JOUFILCLO, EDT$$INTER_ERR
;EXTRN STR$COPY_R
;
;ENTRY EDT$$TERMIN_RCOVMOD, Save R2,R3,R4 : 0720
MOVAB EDT$$G_RCOV_MOD, R4
MOVAB STR$FREE1_DX, R3
MOVAB STR$COPY_R, R2
SUBL2 #20, SP
BLBS EDT$$G_RCOV_MOD, 1$ : 0801
CALLS #0, EDT$$INTER_ERR
MOVL #34471936, FILE_DESC : 0812
MOVL #34471936, RHB_DESC : 0814
CLRL FILE_DESC+4 : 0811
CLRL RHB_DESC+4 : 0813
MOVL #5, RHB_LEN : 0816
PUSHAB P.AAA : 0817
PUSHAB RHB_LEN
PUSHAB RHB_DESC
CALLS #3, STR$COPY_R
BITL EDT$$V_OPTIONS, #EDTSM_NOJOURNAL : 0819
BEQL 2$
PUSHL EDT$$A_JOU_NAM : 0822
PUSHAB EDT$$G_JOU_NAMLEN
PUSHAB FILE_DESC
CALLS #3, STR$COPY_R
CLRL -(SP) : 0823
PUSHAB FILE_DESC
PUSHL #EDT$K_JOURNAL_FILE
PUSHL #EDT$K_CLOSE
CALLS #4, EDT$$CALLFIO
BLBS STATUS, 2$ : 0825
PUSHAB FILE_DESC : 0828
PUSHL #EDT$ JOUFILCLO
CALLS #2, EDT$$FIOPN_ERR
CALLS #0, EDT$$FATAL_IOERR : 0829
CLRL EDT$$G_RCOV_MOD : 0871
CLRL EDT$$G-TIN_OBUFPOS : 0872
PUSHAB FILE_DESC : 0879
CALLS #1, STR$FREE1_DX
PUSHAB RHB_DESC : 0880
CALLS #1, STR$FREE1_DX : 0883
RET

```

; Routine Size: 175 bytes, Routine Base: _EDT\$CODE + 0008

EDTSSAUDIT
V04-000

EDTSSAUDIT - Terminate recovery processing
EDTSS\$TERMIN_RCOVMOD - Terminate recovery proce

1 11
16-Sep-1984 01:28:08
14-Sep-1984 12:24:20

VAX-11 Bliss-32 V4.0-742
DISK\$VM\$MASTER:[EDT.SRC]SAUDIT.BLI;1 Page 8 (3)

: 285
: 286

0884 1
0885 1 !<BLF/PAGE>

EDT
V04

.....
SERIAL C

EDTSSAUDIT
V04-000

EDTSSAUDIT - Terminate recovery processing
EDTSS\$TERMIN_RCOVMOD - Terminate recovery proce

J 11
16-Sep-1984 01:28:08
14-Sep-1984 12:24:20

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]SAUDIT.BLI;1

Page 9
(4)

**f

: 288 0886 1 END
: 289 0887 1
: 290 0888 0 ELUDOM

! of module EDTSSAUDIT

PSECT SUMMARY

Name Bytes Attributes
_EDT\$CODE 183 NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	6	0	581	00:02.6
_\$255\$DUA28:[EDT.SRC]EDT.L32;1	377	3	0	40	00:00.7
_\$255\$DUA28:[EDT.SRC]PSECTS.L32;1	2	1	50	7	00:00.1

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS\$:SAUDIT/OBJ=OBJ\$:SAUDIT MSRC\$:SAUDIT.BLI/UPDATE=(ENH\$:SAUDIT)

: Size: 175 code + 8 data bytes
: Run Time: 00:19.1
: Elapsed Time: 00:28.2
: Lines/CPU Min: 2792
: Lexemes/CPU-Min: 8773
: Memory Used: 101 pages
: Compilation Complete

0138

AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

