


```

WW      WW  FFFFFFFF  GGGGGGGG  EEEEEEEEE  TTTTTTTTT  BBBB8888  KK      KK  TTTTTTTTT
WW      WW  FFFFFFFF  GGGGGGGG  EEEEEEEEE  TTTTTTTTT  BBBB8888  KK      KK  TTTTTTTTT
WW      WW  FF          GG          EE          TT          BB      BB  KK      KK  TT
WW      WW  FF          GG          EE          TT          BB      BB  KK      KK  TT
WW      WW  FF          GG          EE          TT          BB      BB  KK      KK  TT
WW      WW  FF          GG          EE          TT          BB      BB  KK      KK  TT
WW      WW  FFFFFFFF  GG          EE          TT          BBBB8888  KKKKKK  TT
WW      WW  FFFFFFFF  GG          EE          TT          BBBB8888  KKKKKK  TT
WW      WW  FF          GG  GGGGGG  EE          TT          BB      BB  KK      KK  TT
WW      WW  FF          GG  GGGGGG  EE          TT          BB      BB  KK      KK  TT
WWW     WWW  FF          GG          EE          TT          BB      BB  KK      KK  TT
WWW     WWW  FF          GG          EE          TT          BB      BB  KK      KK  TT
WW      WW  FF          GG          EE          TT          BB      BB  KK      KK  TT
WW      WW  FF          GG          EE          TT          BB      BB  KK      KK  TT
WW      WW  FF          GG          EE          TT          BB      BB  KK      KK  TT

```

```

....
....
....
....

```

```

LL      111111  SSSSSSSS
LL      111111  SSSSSSSS
LL      11      SS
LL      11      SS
LL      11      SS
LL      11      SS
LL      11      SSSSSS
LL      11      SSSSSS
LL      11      SS
LL      11      SS
LL      11      SS
LL      11      SS
LLLLLLLL  111111  SSSSSSSS
LLLLLLLL  111111  SSSSSSSS

```

```

1 0001 0 %TITLE 'EDT$WFGETBKT - allocate a bucket'
2 0002 0 MODULE EDT$WFGETBKT ( ! Allocate a bucket
3 0003 0 IDENT = 'V04-000' ! File: WFGETBKT.BLI Edit: JBS1008
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 Allocate a bucket.
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: October 16, 1978
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original. DJS 23-Feb-1981. This module was created by
45 0045 1 extracting routine GET_NEW_BUKT from module EDTWF.
46 0046 1 1-002 - Regularize headers. JBS 16-Mar-1981
47 0047 1 1-003 - Change SY_EXIT to EDT$SYS_EXIT. JBS 31-Mar-1981
48 0048 1 1-004 - Modify to use EDT$WORKIO. STS 15-Feb-1982
49 0049 1 1-005 - Call WF_EXT if running on 11's. STS 26-Feb-1982
50 0050 1 1-006 - Add literals for callable parameters. STS 08-Mar-1982
51 0051 1 1-007 - Fix module name. JBS 07-Apr-1982
52 0052 1 1-008 - Fix work file overflow message. JBS 05-Jul-1982
53 0053 1 --
54 0054 1

```

```
56 0055 1 %SBTTL 'Declarations'  
57 0056 1  
58 0057 1 : TABLE OF CONTENTS:  
59 0058 1 :  
60 0059 1  
61 0060 1 REQUIRE 'EDTSRC:TRAROUNAM';  
62 0499 1  
63 0500 1 FORWARD ROUTINE  
64 0501 1 EDT$WF_ALOBUF : NOVALUE;  
65 0502 1  
66 0503 1 :  
67 0504 1 : INCLUDE FILES:  
68 0505 1 :  
69 0506 1  
70 0507 1 REQUIRE 'EDTSRC:EDTREQ';  
71 0642 1  
72 L 0643 1 %IF %BLISS (BLISS32)  
73 0644 1 %THEN  
74 0645 1  
75 0646 1 REQUIRE 'EDTSRC:SYSSYM';  
76 0676 1  
77 0677 1 %FI  
78 0678 1  
79 0679 1 :  
80 0680 1 : MACROS:  
81 0681 1  
82 0682 1 : NONE  
83 0683 1 :  
84 0684 1 : EQUATED SYMBOLS:  
85 0685 1 :  
86 0686 1  
87 0687 1 EXTERNAL LITERAL  
88 0688 1 EDT$K_GET;  
89 0689 1  
90 0690 1 :  
91 0691 1 : OWN STORAGE:  
92 0692 1 :  
93 0693 1 : NONE  
94 0694 1 :  
95 0695 1 : EXTERNAL REFERENCES:  
96 0696 1 :  
97 0697 1 : In the routine
```

```

: 99      0698 1 %SBTTL 'EDT$$WF_ALOBUF - allocate a bucket'
: 100     0699 1
: 101     0700 1 GLOBAL ROUTINE EDT$$WF_ALOBUF           ! Allocate a bucket
: 102     0701 1   : NOVALUE =
: 103     0702 1
: 104     0703 1   **
: 105     0704 1   FUNCTIONAL DESCRIPTION:
: 106     0705 1
: 107     0706 1       This routine allocates a new bucket from the work-file.  If there
: 108     0707 1       is a bucket available on the deleted bucket list, use it, otherwise
: 109     0708 1       take the next higher numbered bucket.
: 110     0709 1
: 111     0710 1   FORMAL PARAMETERS:
: 112     0711 1
: 113     0712 1       NONE
: 114     0713 1
: 115     0714 1   IMPLICIT INPUTS:
: 116     0715 1
: 117     0716 1       EDT$$G_WK_GRTSTBUK
: 118     0717 1       EDT$$A_WK_BUK
: 119     0718 1
: 120     0719 1   IMPLICIT OUTPUTS:
: 121     0720 1
: 122     0721 1       EDT$$Z_WF_DESC
: 123     0722 1       EDT$$G_WK_AVAIL
: 124     0723 1       EDT$$G_WK_CURBUK
: 125     0724 1       EDT$$G_WK_GRTSTBUK
: 126     0725 1
: 127     0726 1   ROUTINE VALUE:
: 128     0727 1
: 129     0728 1       NONE
: 130     0729 1
: 131     0730 1   SIDE EFFECTS:
: 132     0731 1
: 133     0732 1       If the work file overflows, never returns to its caller.
: 134     0733 1
: 135     0734 1   --
: 136     0735 1
: 137     0736 2   BEGIN
: 138     0737 2
: 139     L 0738 2   %IF %BLISS (BLISS16)
: 140     U 0739 2   %THEN
: 141     U 0740 2
: 142     U 0741 2   EXTERNAL ROUTINE
: 143     U 0742 2   EDT$$WF_EXT;           ! extend the work file by placing a new bucket in the cache
: 144     U 0743 2
: 145     0744 2   %FI
: 146     0745 2
: 147     0746 2   EXTERNAL ROUTINE
: 148     0747 2   EDT$$FMT_MSG;           ! Put the text of a message in the format buffer
: 149     0748 2   EDT$$FMT_CRLF;         ! Terminate the line being built in the format buffer
: 150     0749 2   EDT$$CALWIO;
: 151     0750 2   EDT$$WF_MAKECUR : NOVALUE,
: 152     0751 2   EDT$$SYS_EXI;           ! Leave EDT abruptly
: 153     0752 2
: 154     0753 2   EXTERNAL
: 155     0754 2   EDT$$Z_WF_DESC : BLOCK [, BYTE], ! descriptor for workfile record

```

```

156 0755 2      EDT$$G_WK_AVAIL,           ! Pointer to next available deleted bucket
157 0756      EDT$$G_WK_CURBUK,         ! Number of the current bucket
158 0757      EDT$$G_WK_GRTSTBUK,       ! Largest bucket number in use
159 0758      EDT$$A_WK_BUK :           ! Pointer to current bucket
160 0759      REF B[LOCK [WF_BUKT_SIZE, BYTE] FIELD (WFB_FIELDS);
161 0760
162 0761      MESSAGES ((WRKFILOVF));
163 0762
164 0763      IF (.EDT$$G_WK_AVAIL NEQ 0)
165 0764      THEN
166 0765          BEGIN
167 0766          EDT$$WF_MAKECUR (.EDT$$G_WK_AVAIL);
168 0767          EDT$$G_WK_AVAIL = .EDT$$A_WK_BUK [WFB_NEXT_BUKT];
169 0768          END
170 0769      ELSE
171 0770          BEGIN
172 0771          !+
173 0772          ! Check for overflow
174 0773          !-
175 0774
176 0775          IF ((.EDT$$G_WK_GRTSTBUK EQL 0) OR !
177 0776          (.EDT$$G_WK_GRTSTBUK GTRU 65535))
178 0777          THEN
179 0778              BEGIN
180 0779              EDT$$FMT_MSG (EDT$ WRKFILOVF);
181 0780              EDT$$FMT_CRLF ();
182 0781              EDT$$SYS_EXI (EDT$ WRKFILOVF);
183 0782              END;
184 0783
185 0784          !+
186 0785          ! Inform the caching routines that we are creating a new bucket.
187 0786          ! On the 11's we have to bring another bucket into the cache specially
188 0787          !-
189 0788
190 0789      L  %IF %BLISS (BLISS32)
191 0790      %THEN
192 0791          EDT$$CALLWIO (EDT$K_GET, .EDT$$G_WK_GRTSTBUK, EDT$$Z_WF_DESC);
193 0792      U  %ELSE
194 0793          EDT$$WF_EXT (.EDT$$G_WK_GRTSTBUK, EDT$$A_WK_BUK);
195 0794          EDT$$Z_WF_DESC [DSC$A_POINTER] = .EDT$$A_WK_BUK;
196 0795      U  %FI
197 0796
198 0797          EDT$$A_WK_BUK = .EDT$$Z_WF_DESC [DSC$A_POINTER]; !get address of record
199 0798          EDT$$G_WK_CURBUK = .EDT$$G_WK_GRTSTBUK;
200 0799
201 0800          !+
202 0801          ! And bump the largest bucket number.
203 0802          !-
204 0803          EDT$$G_WK_GRTSTBUK = .EDT$$G_WK_GRTSTBUK + 1;
205 0804          END;
206 0805      END;

```

! End of routine EDT\$\$WF_ALOBUF

```

.TITLE EDT$WFGETBKT EDT$WFGETBKT - allocate a bucket
.IDENT \V04-000\
.EXTRN EDT$K_GET, EDT$$FMT_MSG

```

```

003C 00000
55 00000000G 00 9E 00002
54 00000000G 8F D0 00009
53 00000000G 60 9E 00010
52 00000000G 00 9E 00017
50 00000000G 65 D0 0001E
11 13 00021
50 DD 00023
00000000G 00 01 FB 00025
50 63 D0 0002C
65 02 A0 3C 0002F
04 00033
50 62 D0 00034 1$:
09 13 00037
0000FFFF 8F 50 D1 00039
19 1B 00040
54 DD 00042 2$:
00000000G 00 01 FB 00044
00000000G 00 00 FB 0004B
54 DD 00052
00000000G 00 01 FB 00054
00000000G 00000000G 00 9F 0005B 3$:
62 DD 00061
00000000G 00000000G 8F DD 00063
00000000G 00 03 FB 00069
63 00000000G 00 D0 00070
00000000G 00 62 D0 00077
62 D6 0007E
04 00080

```

```

.ENTRY EDTSSWF_ALOBUF, Save R2,R3,R4,R5 : 0700
MOVAB EDTSSG_WK_AVAIL, R5
MOVL #EDTS_WRKFILOVF, R4
MOVAB EDTSSA_WK_BUK, R3
MOVAB EDTSSG_WK_GRTSTBUK, R2
MOVL EDTSSG_WK_AVAIL, R0 : 0763
BEQL 1$
PUSHL R0 : 0766
CALLS #1, EDTSSWF_MAKECUR
MOVL EDTSSA_WK_BUK, R0 : 0767
MOVZWL 2(R0), EDTSSG_WK_AVAIL
RET : 0763
MOVL EDTSSG_WK_GRTSTBUK, R0 : 0775
BEQL 2$
CML R0, #65535 : 0776
BLEQU 3$
PUSHL R4 : 0779
CALLS #1, EDTSSFMT_MSG
CALLS #0, EDTSSFMT_CRLF : 0780
PUSHL R4 : 0781
CALLS #1, EDTSSSYS_EXI
PUSHAB EDTSSZ_WF_DESC : 0791
PUSHL EDTSSG_WK_GRTSTBUK
PUSHL #EDTSK_GET
CALLS #3, EDTSSCALLWIO
MOVL EDTSSZ_WF_DESC+4, EDTSSA_WK_BUK : 0797
MOVL EDTSSG_WK_GRTSTBUK, EDTSSG_WK_CURBUK : 0798
INCL EDTSSG_WK_GRTSTBUK : 0802
RET : 0805

```

: Routine Size: 129 bytes, Routine Base: _EDTSCODE + 0000

: 207 0806 1
: 208 0807 1 !<BLF/PAGE>

SR
L
C

EDT\$WFGETBKT
V04-000

EDT\$WFGETBKT - allocate a bucket
EDT\$\$WF_ALOBUF - allocate a bucket

J 8
16-Sep-1984 02:06:54
14-Sep-1984 12:25:33

VAX-11 Bliss-32 V4.0-742
DISK\$VMMASTER:[EDT.SRC]WFGETBKT.BLI;1 (4) Page 6

: 210 0808 1 END
: 211 0809 1
: 212 0810 0 ELUDOM

! End of module EDT\$WFGETBKT

PSECT SUMMARY

Name Bytes Attributes
_EDT\$CODE 129 NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(2)

Library Statistics

File	Total	Symbols Loaded	Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[EDT.SRC]EDT.L32:1	377	7	1	40	00:00.2
_\$255\$DUA28:[EDT.SRC]PSECTS.L32:1	2	1	50	7	00:00.1
_\$255\$DUA28:[SYSLIB]STARLET.L32:1	9776	1	0	581	00:04.0

COMMAND QUALIFIERS

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

: Size: 129 code + 0 data bytes
: Run Time: 00:16.9
: Elapsed Time: 00:30.6
: Lines/CPU Min: 2872
: Lexemes/CPU-Min: 7968
: Memory Used: 85 pages
: Compilation Complete

0141 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

The following table lists the LIS components shown in the grid, organized by row:

WMSMSG LIS	WFCOPLIN LIS										
USSTRING LIS										WFCOPY LIS	
			WFDELLIN LIS	WFGETBKT LIS	WFOPNBUF LIS	WFREABCK LIS	WFREAFWD LIS				
											WFSTRINS LIS
	WFAPPBKT LIS										
				WFCVLS LIS					WFCSSC LIS		
UGBUFFER LIS	WFCLEAR LIS										
	USSUBS LIS		WFDEL BKT LIS							WFSPLBKT LIS	
							WFLOCLIN LIS	WFRBKT LIS			
							WFINSLIN LIS		WFREACUR LIS	WFREAINP LIS	WFTOP LIS
										WFREPLIN LIS	
	WFBOTTOM LIS		WFCOPY LIS								