

```

EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTT

```

EXE

Mod

EDT

ED

ED
EDED
EDED
EDED
ED

ED

ED

ED

ED

ED
EDSYN
LBA

110

```
WW      WW      FFFFFFFF      AAAAAA      PPPPPPP      PPPPPPP      BBBB      KK      KK      TTTTTTTTTT
WW      WW      FFFFFFFF      AAAAAA      PPPPPPP      PPPPPPP      BBBB      KK      KK      TTTTTTTTTT
WW      WW      FF      AA      AA      PP      PP      BB      KK      KK      TT
WW      WW      FF      AA      AA      PP      PP      BB      KK      KK      TT
WW      WW      FF      AA      AA      PP      PP      BB      KK      KK      TT
WW      WW      FF      AA      AA      PP      PP      BB      KK      KK      TT
WW      WW      FFFFFFFF      AA      AA      PPPPPPP      PPPPPPP      BBBB      KKKKKK      TT
WW      WW      FFFFFFFF      AA      AA      PPPPPPP      PPPPPPP      BBBB      KKKKKK      TT
WW      WW      FF      AAAAAAAAAA      PP      PP      BB      KK      KK      TT
WW      WW      FF      AAAAAAAAAA      PP      PP      BB      KK      KK      TT
WWWW      WWW      FF      AA      AA      PP      PP      BB      KK      KK      TT
WWWW      WWW      FF      AA      AA      PP      PP      BB      KK      KK      TT
WW      WW      FF      AA      AA      PP      PP      BB      KK      KK      TT
WW      WW      FF      AA      AA      PP      PP      BB      KK      KK      TT
```

```
....
....
....
....
```

```
LL      IIIII      SSSSSSS
LL      IIIII      SSSSSSS
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      IIIII      SSSSSSS
LLLLLLLLLL      IIIII      SSSSSSS
```



```
0001 0 %TITLE 'EDT$WFAPPBKT - append a new bucket'
0002 0 MODULE EDT$WFAPPBKT (
0003 0 IDENT = 'V04-000'
0004 0 ) =
0005 1 BEGIN
0006 1
0007 1 *****
0008 1 *
0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0011 1 * ALL RIGHTS RESERVED.
0012 1 *
0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0018 1 * TRANSFERRED.
0019 1 *
0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0022 1 * CORPORATION.
0023 1 *
0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0026 1 *
0027 1 *
0028 1 *****
0029 1
0030 1
0031 1 ++
0032 1 FACILITY: EDT -- The DEC Standard Editor
0033 1
0034 1 ABSTRACT:
0035 1
0036 1 Append a new bucket to a text buffer.
0037 1
0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
0039 1
0040 1 AUTHOR: Bob Kushlis, CREATION DATE: October 16, 1978
0041 1
0042 1 MODIFIED BY:
0043 1
0044 1 1-001 - Original. DJS 23-Feb-1981. This module was created by
0045 1 extracting routine APPEND_BUKT from module EDTWF.
0046 1 1-002 - Regularize headers. JBS 16-Mar-1981
0047 1 1-003 - Modify to use edt$workio. STS 15-Feb-1982
0048 1 1-004 - Add literals for callable parameters. STS 08-Mar-1982
0049 1 1-005 - Put wf_nxt_buk in line. STS 11-Oct-1982
0050 1 1-006 - Improve the appearance of the listing. JBS 20-Jun-1983
0051 1 --
0052 1
```

EDT\$WFAPPBKT
V04-000

EDT\$WFAPPBKT - append a new bucket
Declarations

H 3
16-Sep-1984 02:01:49
14-Sep-1984 12:25:26

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]WFAPPBKT.BLI;1 Page 2 (2)

```

: 54      0053 1 %SBTTL 'Declarations'
: 55      0054 1
: 56      0055 1 : TABLE OF CONTENTS:
: 57      0056 1 :
: 58      0057 1
: 59      0058 1 REQUIRE 'EDT$SRC:TRAROUNAM';
: 60      0497 1
: 61      0498 1 FORWARD ROUTINE
: 62      0499 1     EDT$WF_NEWBUK : NOVALUE;
: 63      0500 1
: 64      0501 1 :
: 65      0502 1 : INCLUDE FILES:
: 66      0503 1 :
: 67      0504 1
: 68      0505 1 REQUIRE 'EDT$SRC:EDTREQ';
: 69      0640 1
: 70      0641 1 :
: 71      0642 1 : MACROS:
: 72      0643 1 :
: 73      0644 1 :     NONE
: 74      0645 1 :
: 75      0646 1 : EQUATED SYMBOLS:
: 76      0647 1 :
: 77      0648 1
: 78      0649 1 EXTERNAL LITERAL
: 79      0650 1     EDT$K_PUT;
: 80      0651 1
: 81      0652 1 :
: 82      0653 1 : OWN STORAGE:
: 83      0654 1 :
: 84      0655 1 :     NONE
: 85      0656 1 :
: 86      0657 1 : EXTERNAL REFERENCES:
: 87      0658 1 :
: 88      0659 1 :     In the routine

```

**F


```

90 0660 1 %SBTTL 'EDT$$WF_NEWBUK - append a new bucket'
91 0661 1
92 0662 1 GLOBAL ROUTINE EDT$$WF_NEWBUK (
93 0663 1     NEXT,
94 0664 1     PREV
95 0665 1     ) : NOVALUE =
96 0666 1
97 0667 1 ++
98 0668 1 FUNCTIONAL DESCRIPTION:
99 0669 1
100 0670 1     This routine appends a new bucket to a text buffer. The bucket is
101 0671 1     linked into the chain.
102 0672 1
103 0673 1 FORMAL PARAMETERS:
104 0674 1
105 0675 1     NEXT                the bucket which will follow the new bucket or 0 if it is at the end.
106 0676 1
107 0677 1     PREV                the bucket which will precede the new bucket or 0 if it is the first.
108 0678 1
109 0679 1 IMPLICIT INPUTS:
110 0680 1
111 0681 1     EDT$$A_CUR_BUF
112 0682 1     EDT$$A_WK_BUK
113 0683 1     EDT$$G_WK_CURBUK
114 0684 1     EDT$$A_WK_LN
115 0685 1     EDT$$Z_WF_DESC
116 0686 1     EDT$$G_WK_MODFD
117 0687 1     EDT$$G_WK_AVAIL
118 0688 1     EDT$$G_WK_GTRSTBUK
119 0689 1
120 0690 1 IMPLICIT OUTPUTS:
121 0691 1
122 0692 1     EDT$$A_CUR_BUF
123 0693 1     EDT$$G_WK_MODFD
124 0694 1     EDT$$G_WK_CURBUK
125 0695 1
126 0696 1 ROUTINE VALUE:
127 0697 1
128 0698 1     NONE
129 0699 1
130 0700 1 SIDE EFFECTS:
131 0701 1
132 0702 1     NONE
133 0703 1
134 0704 1 --
135 0705 1
136 0706 2 BEGIN
137 0707 2
138 0708 2 EXTERNAL ROUTINE
139 0709 2     EDT$$CALLWIO,
140 0710 2     EDT$$WF_ALOBUF : NOVALUE,
141 0711 2     EDT$$WF_MAKECUR : NOVALUE;
142 0712 2
143 0713 2 EXTERNAL
144 0714 2     EDT$$Z_WF_DESC : BLOCK [8, BYTE],
145 0715 2     EDT$$A_CUR_BUF : REF TBCB_BLOCK,
146 0716 2     EDT$$G_WK_AVAIL,
```

```

! Append a new bucket to a text buffer
! Next bucket, or 0
! Previous bucket, or 0
```

```
! calls appropriate workfile routine
```

```
! workfile record descriptor
! Current text buffer control block
```



```
147 0717 2 EDT$$G_WK_GRTSTBUK,  
148 0718 2 EDT$$A_WK_BUK : ! Pointer to current bucket  
149 0719 2 REF BLOCK [WF_BUKT_SIZE, BYTE] FIELD (WFB_FIELDS),  
150 0720 2 EDT$$G_WK_CURBUK, ! Number of the current bucket  
151 0721 2 EDT$$A_WK_LN : REF LIN_BLOCK, ! Pointer to current line  
152 0722 2 EDT$$G_WK_MODFD; ! Flag indicating bucket was modified  
153 0723 2  
154 0724 2  
155 0725 2 !+ Is this the last bucket in the buffer?  
156 0726 2 !-  
157 0727 2  
158 0728 2 IF (.NEXT EQL 0)  
159 0729 2 THEN  
160 0730 2 !+  
161 0731 2 Yes, update the last bucket field of EDT$$A_CUR_BUF  
162 0732 2 !-  
163 0733 2  
164 0734 2 IF (.EDT$$G_WK_AVAIL NEQ 0)  
165 0735 2 THEN  
166 0736 2 EDT$$A_CUR_BUF [TBCB_LAST_BUKT] = .EDT$$G_WK_AVAIL  
167 0737 2 ELSE  
168 0738 2 EDT$$A_CUR_BUF [TBCB_LAST_BUKT] = .EDT$$G_WK_GRTSTBUK  
169 0739 2  
170 0740 2 ELSE  
171 0741 2 !+  
172 0742 2 No, link the next bucket back to the one we are adding.  
173 0743 2 !-  
174 0744 2 BEGIN  
175 0745 2 EDT$$WF_MAKECUR (.NEXT);  
176 0746 2  
177 0747 2 IF (.EDT$$G_WK_AVAIL NEQ 0)  
178 0748 2 THEN  
179 0749 2 EDT$$A_WK_BUK [WFB_PREV_BUKT] = .EDT$$G_WK_AVAIL  
180 0750 2 ELSE  
181 0751 2 EDT$$A_WK_BUK [WFB_PREV_BUKT] = .EDT$$G_WK_GRTSTBUK;  
182 0752 2  
183 0753 2 EDT$$G_WK_MODFD = 1;  
184 0754 2 END;  
185 0755 2  
186 0756 2 !+  
187 0757 2 Write out the current bucket if it has been modified.  
188 0758 2 !-  
189 0759 2  
190 0760 2 IF .EDT$$G_WK_MODFD THEN EDT$$CALLWIO (EDT$K_PUT, .EDT$$G_WK_CURBUK, EDT$$Z_WF_DESC);  
191 0761 2  
192 0762 2 EDT$$G_WK_MODFD = 0;  
193 0763 2 !+  
194 0764 2 Get a new bukt.  
195 0765 2 !-  
196 0766 2 EDT$$WF_ALOBUF ();  
197 0767 2 !+  
198 0768 2 Update the current bucket and next bucket info  
199 0769 2 !-  
200 0770 2 EDT$$A_CUR_BUF [TBCB_CUR_BUKT] = .EDT$$G_WK_CURBUK;  
201 0771 2 !+  
202 0772 2 Fill in the bucket info for the new bucket  
203 0773 2 !-
```

EDT\$WFAPPBKT
V04-000

EDT\$WFAPPBKT - append a new bucket
EDT\$\$WF_NEWBUK - append a new bucket

K 3
16-Sep-1984 02:01:49
14-Sep-1984 12:25:26

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]WFAPPBKT.BLI;1 Page 5
(3)

```

204 0774 2 EDT$$A_WK_BUK [WFB_NEXT_BUKT] = .NEXT;
205 0775 2 EDT$$A_WK_BUK [WFB_PREV_BUKT] = .PREV;
206 0776 2 EDT$$A_WK_BUK [WFB_END] = WFB_FIXED_SIZE;
207 0777 2
208 0778 2 + Update EDT$$A_CUR_BUF to point to first record in this new bucket
209 0779 2 -
210 0780 2 EDT$$A_CUR_BUF [TBCB_LINE_ADDR] = WFB_FIXED_SIZE;
211 0781 2 EDT$$A_WK_LN = CH$PTR (.EDT$$A_WK_BUK, WFB_FIXED_SIZE);
212 0782 2
213 0783 2 + Mark the new bucket as modified.
214 0784 2 -
215 0785 2 EDT$$G_WK_MODFD = 1;
216 0786 1 END;
```

! of routine EDT\$\$WF_NEWBUK

.TITLE EDT\$WFAPPBKT EDT\$WFAPPBKT - append a new bucket
.IDENT \V04-000\

.EXTRN EDT\$K_PUT, EDT\$\$CALLWIO
.EXTRN EDT\$\$WF_ALOBUF, EDT\$\$WF_MAKECUR
.EXTRN EDT\$\$Z_WF_DESC, EDT\$\$A_CUR_BUF
.EXTRN EDT\$\$G_WK_AVAIL
.EXTRN EDT\$\$G_WK_GRTSTBUK
.EXTRN EDT\$\$A_WK_BUK, EDT\$\$G_WK_CURBUK
.EXTRN EDT\$\$A_WK_LN, EDT\$\$G_WK_MODFD

.PSECT _EDT\$CODE, NOWRT, SHR, PIC, 2

			01FC 00000	.ENTRY	EDT\$\$WF_NEWBUK, Save R2,R3,R4,R5,R6,R7,R8	0662
58	00000000G	00	9E 00002	MOVAB	EDT\$\$A_CUR_BUF, R8	
57	00000000G	00	9E 00009	MOVAB	EDT\$\$G_WK_CURBUK, R7	
56	00000000G	00	9E 00010	MOVAB	EDT\$\$A_WK_BUK, R6	
55	00000000G	00	9E 00017	MOVAB	EDT\$\$G_WK_GRTSTBUK, R5	
54	00000000G	00	9E 0001E	MOVAB	EDT\$\$G_WK_AVAIL, R4	
53	00000000G	00	9E 00025	MOVAB	EDT\$\$G_WK_MODFD, R3	
52	04	AC	D0 0002C	MOVL	NEXT, R2	0728
		14	12 00030	BNEQ	2\$	
50		68	D0 00032	MOVL	EDT\$\$A_CUR_BUF, R0	0736
51		64	D0 00035	MOVL	EDT\$\$G_WK_AVAIL, R1	0734
		06	13 00038	BEQL	1\$	
10	A0	51	B0 0003A	MOVW	R1, 16(R0)	0736
		22	11 0003E	BRB	5\$	
10	A0	65	B0 00040	MOVW	EDT\$\$G_WK_GRTSTBUK, 16(R0)	0738
		1C	11 00044	BRB	5\$	0734
		52	DD 00046	PUSHL	R2	0745
	00000000G	00	01 FB 00048	CALLS	#1, EDT\$\$WF_MAKECUR	
51		66	D0 0004F	MOVL	EDT\$\$A_WK_BUK, R1	0749
50		64	D0 00052	MOVL	EDT\$\$G_WK_AVAIL, R0	0747
		05	13 00055	BEQL	3\$	
61		50	B0 00057	MOVW	R0, (R1)	0749
		03	11 0005A	BRB	4\$	
61		65	B0 0005C	MOVW	EDT\$\$G_WK_GRTSTBUK, (R1)	0751
63		01	D0 0005F	MOVL	#1, EDT\$\$G_WK_MODFD	0753
15		63	E9 00062	BLBC	EDT\$\$G_WK_MODFD, 6\$	0760
	00000000G	00	9F 00065	PUSHAB	EDT\$\$Z_WF_DESC	
		67	DD 0006B	PUSHL	EDT\$\$G_WK_CURBUK	
	00000000G	8F	DD 0006D	PUSHL	#EDT\$K_PUT	

EDT\$WFAPPBKT
V04-000

EDT\$WFAPPBKT - append a new bucket
EDT\$\$WF_NEWBUK - append a new bucket

L 3
16-Sep-1984 02:01:49
14-Sep-1984 12:25:26

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]WFAPPBKT.BLI;1
Page 6 (3)

00000000G	00	03	FB	00073	CALLS	#3, EDT\$\$CALLWIO	:	
		63	D4	0007A	CLRL	EDT\$\$G_WK_MODFD	:	0762
00000000G	00	00	FB	0007C	CALLS	#0, EDT\$\$WF_ALOBUF	:	0766
	51	68	D0	00083	MOVL	EDT\$\$A_CUR_BUF, R1	:	0770
04	A1	67	B0	00086	MOVW	EDT\$\$G_WK_CURBUK, 4(R1)	:	
	50	66	D0	0008A	MOVL	EDT\$\$A_WK_BUK, R0	:	0774
02	A0	52	B0	0008D	MOVW	R2, 2(R0)	:	
	60	08	AC	B0 00091	MOVW	PREV, (R0)	:	0775
04	A0	08	D0	00095	MOVL	#8, 4(R0)	:	0776
	61	08	D0	00099	MOVL	#8, (R1)	:	0780
00000000G	00	08	A0	9E 0009C	MOVAB	8(R0), EDT\$\$A_WK_LN	:	0781
	63	01	D0	000A4	MOVL	#1, EDT\$\$G_WK_MODFD	:	0785
		04	000A7	RET			:	0786

; Routine Size: 168 bytes, Routine Base: _EDT\$CODE + 0000

: 217 0787 1
: 218 0788 1 !<BLF/PAGE>

EDT\$WFAPPBKT
V04-000

EDT\$WFAPPBKT - append a new bucket
EDT\$WF_NEWBUK - append a new bucket

M 3
16-Sep-1984 02:01:49
14-Sep-1984 12:25:26

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

: 220 0789 1 END
: 221 0790 1
: 222 C791 0 ELUDOM

! of module EDT\$WFAPPBKT

PSECT SUMMARY

: Name Bytes Attributes
: _EDT\$CODE 168 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 39 10 40 00:00.2
: _\$255\$DUA28:[EDT.SRC]PSECTS.L32;1 2 1 50 7 00:00.1

COMMAND QUALIFIERS

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

: Size: 168 code + 0 data bytes
: Run Time: 00:13.4
: Elapsed Time: 00:18.2
: Lines/CPU Min: 3547
: Lexemes/CPU-Min: 10569
: Memory Used: 86 pages
: Compilation Complete

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

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

