```
EEEEEEEEEEEEEEEE   DDDDDDDDDD      TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD      TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD      TTTTTTTTTTTTTTTT
EEE                DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEEEEEEEEEEE       DDD       DDD         TTT
EEEEEEEEEEEE       DDD       DDD         TTT
EEEEEEEEEEEE       DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEE                DDD       DDD         TTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD            TTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD            TTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD            TTT
```

FILEID**WFGETBKT

WFGETBKT
LIS

```
    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_EXI.  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
```

EDT$WFGETBKT   EDT$WFGETBKT - allocate a bucket   F 8
V04-000   Declarations   16-Sep-1984 02:06:54   VAX-11 Bliss-32 V4.0-742   Page 2
                                                 14-Sep-1984 12:25:33   DISK$VMSMASTER:[EDT.SRC]WFGETBKT.BLI;1 (2)

EDT$WFGETBKT   EDT$WFGETBKT - allocate a bucket   G 8
V04-000   EDT$$WF_ALOBUF - allocate a bucket   16-Sep-1984 02:06:54   VAX-11 Bliss-32 V4.0-742   Page 3
                                                                 14-Sep-1984 12:25:33   DISK$VMSMASTER:[EDT.SRC]WFGETBKT.BLI;1   (3)

```
  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$$FMT_CRLF,                           ! Terminate the line being built in the format buffer
 150     0749   2           EDT$$CAL_WIO,
 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$$G_WK_AVAIL,                              ! Pointer to next available deleted bucket
 157    0756   2        EDT$$G_WK_CURBUK,                             ! Number of the current bucket
 158    0757   2        EDT$$G_WK_GRTSTBUK,                           ! Largest bucket number in use
 159    0758   2        EDT$$A_WK_BUK :                               ! Pointer to current bucket
 160    0759   2            REF BLOCK [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_QK_AVAIL = .EDT$$A_WR_BUK [WFB_NEXT_BUKT];
 169    0768              END
 170    0769          ELSE
 171    0770              BEGIN
 172    0771   !+
 173    0772   ! Check for overflow
 174    0773   !-
 175    0774
 176    0775   4          IF ((.EDT$$G_WK_GRTSTBUK EQL 0) OR           !
 177    0776   4              (.EDT$$G_WK_GRTSTBUK GTRU 65535))
 178    0777   3          THEN
 179    0778   4              BEGIN
 180    0779   4              EDT$$FMT_MSG (EDT$_WRKFILOVF);
 181    0780   4              EDT$$FMT_CRLF ();
 182    0781   4              EDT$$SYS_EXI (EDT$_WRKFILOVF);
 183    0782   3              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  L 0789   3  %IF %BLISS (BLISS32)
 191    0790   3  %THEN
 192    0791   3          EDT$$CALLWIO (EDT$K_GET, .EDT$$G_WK_GRTSTBUK, EDT$$Z_WF_DESC);
 193    0792   3  %ELSE
 194  U 0793   3          EDT$$WF_EXT (.EDT$$G_WK_GRTSTBUK, EDT$$A_WK_BUK);
 195  U 0794   3          EDT$$Z_QF_DESC [DSC$A_POINTER] = .EDT$$A_WK_BUK;
 196  U 0795   3  %FI
 197    0796
 198    0797   3          EDT$$A_WK_BUK = .EDT$$Z_WF_DESC [DSC$A_POINTER];       !get address of record
 199    0798   3          EDT$$G_WK_CURBUK = .EDT$$G_WK_GRTSTBUK;
 200    0799   !+
 201    0800   ! And bump the largest bucket number.
 202    0801   !-
 203    0802   2          EDT$$G_WK_GRTSTBUK = .EDT$$G_WK_GRTSTBUK + 1;
 204    0803   2          END;
 205    0804
 206    0805   1      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
```

EDT$WFGETBKT   EDT$WFGETBKT - allocate a bucket   16-Sep-1984 02:06:54   VAX-11 Bliss-32 V4.0-742   Page   5
V04-000   EDT$$WF_ALOBUF - allocate a bucket   14-Sep-1984 12:25:33   DISK$VMSMASTER:[EDT.SRC]WFGETBKT.BLI;1 (3)

I 8

```
                                                                    .EXTRN    EDT$$FMT_CRLF, EDT$$CALLWIO
                                                                    .EXTRN    EDT$$WF_MAKECUR
                                                                    .EXTRN    EDT$$SYS_EXI, EDT$$Z_WF_DESC
                                                                    .EXTRN    EDT$$G_WK_AVAIL
                                                                    .EXTRN    EDT$$G_WK_CURBUK
                                                                    .EXTRN    EDT$$G_WK_GRTSTBUK
                                                                    .EXTRN    EDT$$A_WK_BUK, EDT$_WRKFILOVF

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

                                         003C 00000                 .ENTRY    EDT$$WF_ALOBUF, Save R2,R3,R4,R5          ; 0700
                        55 00000000G  00   9E 00002                 MOVAB     EDT$$G_QK_AVAIL, R5
                        54 00000000G  8F   D0 00009                 MOVL      #EDT$_QRKFILOVF, R4
                        53 00000000G  00   9E 00010                 MOVAB     EDT$$A_WK_BUK, R3
                        52 00000000G  00   9E 00017                 MOVAB     EDT$$G_WK_GRTSTBUK, R2
                        50           65   D0 0001E                  MOVL      EDT$$G_WK_AVAIL, R0              ; 0763
                        11           13 00021                       BEQL      1$                               ; 0763
                        50           DD 00023                       PUSHL     R0                               ; 0766
         00000000G  00  01   FB 00025                               CALLS     #1, EDT$$WF_MAKECUR
                        50           63   D0 0002C                  MOVL      EDT$$A_WK_BUK, R0                ; 0767
                        65        02 A0   3C 0002F                  MOVZWL    2(R0), -EDT$$G_WK_AVAIL
                        04 00033                                    RET                                       ; 0763
                        50           62   D0 00034 1$:              MOVL      EDT$$G_WK_GRTSTBUK, R0           ; 0775
                        09           13 00037                       BEQL      2$
         0000FFFF  8F  50   D1 00039                                CMPL      R0, #65535                       ; 0776
                        19           1B 00040                       BLEQU     3$
                        54           DD 00042 2$:                   PUSHL     R4                               ; 0779
         00000000G  00  01   FB 00044                               CALLS     #1, EDT$$FMT_MSG
         00000000G  00  00   FB 0004B                               CALLS     #0, EDT$$FMT_CRLF                ; 0780
                        54           DD 00052                       PUSHL     R4                               ; 0781
         00000000G  00  01   FB 00054                               CALLS     #1, EDT$$SYS_EXI
         00000000G  00   9F 0005B 3$:                               PUSHAB    EDT$$Z_WF_DESC                   ; 0791
                        62           DD 00061                       PUSHL     EDT$$G_WK_GRTSTBUK
         00000000G  8F   DD 00063                                   PUSHL     #EDT$K_GET
         00000000G  00  03   FB 00069                               CALLS     #3, EDT$$CALLWIO
         00000000G  00  63 00000000G  00   D0 00070                 MOVL      EDT$$Z_WF_DESC+4, EDT$$A_WK_BUK  ; 0797
         00000000G  00  62   D0 00077                               MOVL      EDT$$G_WK_GRTSTBUK, EDT$$G_QK_CURBUK  ; 0798
                        62           D6 0007E                       INCL      EDT$$G_WK_GRTSTBUK               ; 0802
                        04 00080                                    RET                                       ; 0805
```

; Routine Size: 129 bytes,   Routine Base: _EDT$CODE + 0000

```
;  207           0806  1
;  208           0807  1 !<BLF/PAGE>
```

```
; 210          0808  1 END                                              ! End of module EDT$WFGETBKT
; 211          0809  1
; 212          0810  0 ELUDOM
```


                              PSECT SUMMARY

        Name   Bytes   Attributes

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

                          Library Statistics

                                -------- Symbols --------   Pages   Processing
        File   Total   Loaded   Percent   Mapped   Time

     _255DUA28:[EDT.SRC]EDT.L32;1   377   7   1   40   00:00.2
     _255DUA28:[EDT.SRC]PSECTS.L32;1   2   1   50   7   00:00.1
     _255DUA28:[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$:WFGETBKT.BLI/UPDATE=(ENH$:WFGE
        TBKT)

; 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

VMSMSG
LIS

WFCOPLIN
LIS

USSTRING
LIS

WFSCOPY
LIS

WFDELLIN
LIS

WFGETBKT
LIS

WFOPNBUF
LIS

WFREABCK
LIS

WFREAFWD
LIS

WFSTRINS
LIS

WFAPPBKT
LIS

WFSNDNS
LIS

WFRESSB
LIS

USBUFFER
LIS

WFCLEAR
LIS

USSUBS
LIS

WFDELBKT
LIS

WFSPLBKT
LIS

WFLOCLIN
LIS

WFRBUKT
LIS

WFINSLIN
LIS

WFREACUR
LIS

WFREAINP
LIS

WFTOP
LIS

WFREPLIN
LIS

WFBOTTOM
LIS

WFECOPY
LIS