


```

WW      WW  FFFFFFFF  RRRRRRRR  EEEEEEEEE  SSSSSSSS  EEEEEEEEE  QQQQQQ
WW      WW  FFFFFFFF  RRRRRRRR  EEEEEEEEE  SSSSSSSS  EEEEEEEEE  QQQQQQ
WW      WW  FF          RR          RR  EE          SS          EE          QQ          QQ
WW      WW  FF          RR          RR  EE          SS          EE          QQ          QQ
WW      WW  FF          RR          RR  EE          SS          EE          QQ          QQ
WW      WW  FF          RR          RR  EE          SS          EE          QQ          QQ
WW      WW  FFFFFFFF  RRRRRRRR  EEEEEEEEE  SSSSSSSS  EEEEEEEEE  QQ          QQ
WW      WW  FFFFFFFF  RRRRRRRR  EEEEEEEEE  SSSSSSSS  EEEEEEEEE  QQ          QQ
WW  WW  WW  FF          RR  RR          EE          SS          EE          QQ  QQ  QQ
WW  WW  WW  FF          RR  RR          EE          SS          EE          QQ  QQ  QQ
WWW  WWW  FF          RR  RR          EE          SS          EE          QQ  QQ  QQ
WWW  WWW  FF          RR  RR          EE          SS          EE          QQ  QQ  QQ
WW      WW  FF          RR          RR  EEEEEEEEE  SSSSSSSS  EEEEEEEEE  QQQQ  QQ
WW      WW  FF          RR          RR  EEEEEEEEE  SSSSSSSS  EEEEEEEEE  QQQQ  QQ

```

```

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

```

```

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
LLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLL  IIIIII  SSSSSSSS

```

```

: R
:
:
:

```

```

1 0001 0 XTITLE 'EDT$WFRESEQ - resequence a range of lines'
2 0002 0 MODULE EDT$WFRESEQ ( ! Resequence a range of lines
3 0003 0 IDENT = 'V04-000' ! File: WFRESEQ.BLI Edit: STS1011
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 Resequence a range of lines.
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 EDT$RSEQ from module EDTWF.
46 0046 1 1-002 - Regularize headers. JBS 19-Mar-1981
47 0047 1 1-003 - Add return value and line_too_large error check. SMB 4-Feb-1982
48 0048 1 1-004 - Change count to 48-bits. SMB 07-Feb-1982
49 0049 1 1-005 - Only copy edt$a_cur_buf once. STS 18-Mar-1982
50 0050 1 1-006 - Fix subtitle and local symbol name. JBS 14-Sep-1982
51 0051 1 1-007 - Put edt$stst_eob in line. STS 22-Sep-1982
52 0052 1 1-008 - Declare EDT$Z EOB_LN. JBS 23-Sep-1982
53 0053 1 1-009 - Remove declaration of EDT$STST EOB, no longer used. JBS 24-Sep-1982
54 0054 1 1-010 - Modify to use new 48 bit macros. STS 01-Oct-1982
55 0055 1 1-011 - Modify to use new compare macro. STS 20-Oct-1982
56 0056 1 --
57 0057 1

```

EDT\$WFRESEQ
V04-000

EDT\$WFRESEQ - resequence a range of Lines
Declarations

I 14
16-Sep-1984 02:13:45
14-Sep-1984 12:25:43

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

**F

```
: 59 0058 1 %SBTTL 'Declarations'  
: 60 0059 1  
: 61 0060 1 !! TABLE OF CONTENTS:  
: 62 0061 1 !  
: 63 0062 1  
: 64 0063 1 REQUIRE 'EDT$SRC:TRAROUNAM';  
: 65 0502 1  
: 66 0503 1 FORWARD ROUTINE  
: 67 0504 1 EDT$SRSEQ;  
: 68 0505 1  
: 69 0506 1 !  
: 70 0507 1 !! INCLUDE FILES:  
: 71 0508 1 !  
: 72 0509 1  
: 73 0510 1 REQUIRE 'EDT$SRC:EDTREQ';  
: 74 0645 1  
: 75 0646 1 !  
: 76 0647 1 !! MACROS:  
: 77 0648 1  
: 78 0649 1 NONE  
: 79 0650 1  
: 80 0651 1 !! EQUATED SYMBOLS:  
: 81 0652 1  
: 82 0653 1 NONE  
: 83 0654 1  
: 84 0655 1 !! OWN STORAGE:  
: 85 0656 1  
: 86 0657 1 NONE  
: 87 0658 1  
: 88 0659 1 !! EXTERNAL REFERENCES:  
: 89 0660 1  
: 90 0661 1 ! In the routine
```

```

: 92 0662 1 %SBTTL 'EDT$$RSEQ - resequence a range of lines'
: 93 0663 1
: 94 0664 1 GLOBAL ROUTINE EDT$$RSEQ (           ! Resequence a range of lines
: 95 0665 1     COUNT,                          ! Number of lines
: 96 0666 1     START,                          ! Initial line number
: 97 0667 1     INC                             ! Increment
: 98 0668 1     ) =
: 99 0669 1
100 0670 1 +-+
101 0671 1  FUNCTIONAL DESCRIPTION:
102 0672 1
103 0673 1     This routine resequences a range of lines. The current line is
104 0674 1     the first to be resequenced. The line following the last line
105 0675 1     resequenced becomes the current line. This routine works by assigning
106 0676 1     the number START to the first line and adding INC to get the next line
107 0677 1     number. If when we have finished, the number of the following line is
108 0678 1     less than the last one resequenced, then we continue resequencing until
109 0679 1     this is no longer true.
110 0680 1
111 0681 1  FORMAL PARAMETERS:
112 0682 1
113 0683 1     COUNT                          the number of lines to be resequenced
114 0684 1
115 0685 1     START                          the initial line number
116 0686 1
117 0687 1     INC                            the line number increment.
118 0688 1
119 0689 1  IMPLICIT INPUTS:
120 0690 1
121 0691 1     EDT$$A_CUR_BUF
122 0692 1     EDT$$A_WK_CN
123 0693 1
124 0694 1  IMPLICIT OUTPUTS:
125 0695 1
126 0696 1     EDT$$A_CUR_BUF
127 0697 1     EDT$$G_WK_MODFD
128 0698 1
129 0699 1  ROUTINE VALUE:
130 0700 1
131 0701 1     0 = unable to resequence some of the lines
132 0702 1     1 = successful resequence
133 0703 1
134 0704 1  SIDE EFFECTS:
135 0705 1
136 0706 1     NONE
137 0707 1
138 0708 1  --
139 0709 1
140 0710 2  BEGIN
141 0711 2
142 0712 2  EXTERNAL ROUTINE
143 0713 2     EDT$$FMT_MSG,
144 0714 2     EDT$$RD_CURLN : NOVALUE,
145 0715 2     EDT$$RD_NXTLN;
146 0716 2
147 0717 2  EXTERNAL
148 0718 2     EDT$$L_WK_RESCNT : LN_BLOCK,           ! Actual # lines resequenced

```

```

149 0719 2      EDTSSL_LNO0 : LN_BLOCK,
150 0720 2      EDTSSL_LNO_ZERO : LN_BLOCK,
151 0721 2      EDTSSL_LNO_BIG,
152 0722 2      EDTSSA_CUR_BUF : REF TBFB_BLOCK,
153 0723 2      EDTSSA_WK_LN : REF LIN_BLOCK,
154 0724 2      EDTSSG_WK_MODFD,
155 0725 2      EDTSSZ_EOB_LN : REF LIN_BLOCK;
156 0726 2
157 0727 2      MESSAGES ((MAXLINNUM));
158 0728 2
159 0729 2      LOCAL
160 0730 2      COPY_FLAG,
161 0731 2      EOF,
162 0732 2      MAX,
163 0733 2      LINE_TOO_LARGE,
164 0734 2      SAV_BUFPOS : POS_BLOCK,
165 0735 2      LINE_CNT : LN_BLOCK,
166 0736 2      NEXT_LINE_NUMBER : LN_BLOCK,
167 0737 2      PREV_NUM : LN_BLOCK;
168 0738 2
169 0739 2      MOVELINE (.COUNT, LINE_CNT);
170 0740 2      MOVELINE (.COUNT, EDTSSL_WK_RESCNT);
171 0741 2      COPY_FLAG = 0;
172 0742 2
173 0743 2      IF (LINNOEQL (LINE_CNT, EDTSSL_LNO_ZERO) ) THEN RETURN (0);
174 0744 2
175 0745 2      EOF = 0;
176 0746 2      MOVELINE (.START, NEXT_LINE_NUMBER);
177 0747 2      MOVELINE (.START, PREV_NUM);
178 0748 2      LINE_TOO_LARGE = 0;
179 0749 2
180 0750 2      WHILE ( NOT .EOF) DO
181 0751 2          BEGIN
182 0752 2
183 0753 3      !+ Determine if there are more lines to resequence than what was specified
184 0754 3      ! in COUNT. If so, increase the "actual resequence count".
185 0755 3      !-
186 0756 3
187 0757 4      IF (LINNOEQL (LINE_CNT, EDTSSL_LNO_ZERO) )
188 0758 3      THEN
189 0759 3
190 0760 4      IF (CMLNO (EDTSSA_WK_LN [LIN_NUM], PREV_NUM) GTR 0)
191 0761 3      THEN
192 0762 3          EXITLOOP
193 0763 3      ELSE
194 0764 4          ADDLINE (NUMBER_ONE, EDTSSL_WK_RESCNT)
195 0765 3
196 0766 3      ELSE
197 0767 3          SUBLINE (NUMBER_ONE, LINE_CNT);
198 0768 3
199 0769 3      MOVELINE (NEXT_LINE_NUMBER, EDTSSA_WK_LN [LIN_NUM]);
200 0770 3      EDTSSG_WK_MODFD = 1;
201 0771 3      EDTSSRD_NXTLN ();
202 0772 3
203 0773 4      IF (LINNOEQL (LINE_CNT, EDTSSL_LNO_ZERO) )
204 0774 3      THEN
205 0775 3

```

```

206 0776 4          IF ( NOT .COPY_FLAG)
207 0777 3          THEN
208 0778 4          BEGIN
209 0779 4          EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, SAV_BUFPOS);
210 0780 4          COPY_FLAG = 1;
211 0781 3          END;
212 0782 3
213 0783 3          MOVELINE (NEXT_LINE_NUMBER, PREV_NUM);
214 0784 3
215 0785 4          IF (.EDT$$A_WK_LN NEQA EDT$$Z_EOB_LN)
216 0786 3          THEN
217 0787 4          BEGIN
218 0788 4          ADDLINE (.INC, NEXT_LINE_NUMBER, NEXT_LINE_NUMBER, MAX);
219 0789 4
220 0790 5          IF ((CMLPNO (NEXT_LINE_NUMBER, EDT$$L_LNO_BIG) GTR 0) OR (.MAX NEQ 0))
221 0791 4          THEN
222 0792 5          BEGIN
223 0793 5          LINE_TOO_LARGE = 1;
224 0794 6          MOVELINE (EDT$$L_LNO_BIG, NEXT_LINE_NUMBER)
225 0795 4          END;
226 0796 4
227 0797 4          END
228 0798 3          ELSE
229 0799 3          EOF = 1;
230 0800 3
231 0801 2          END;
232 0802 2
233 0803 2          EDT$$CPY_MEM (POS_SIZE, SAV_BUFPOS, .EDT$$A_CUR_BUF);
234 0804 2          EDT$$RD_CURLN ();
235 0805 2
236 0806 3          IF (.LINE_TOO_LARGE NEQ 0)
237 0807 2          THEN
238 0808 3          BEGIN
239 0809 3          EDT$$FMT_MSG (EDT$_MAXLINNUM);
240 0810 4          RETURN (0)
241 0811 3          END
242 0812 2          ELSE
243 0813 3          RETURN (1)
244 0814 3
245 0815 1          END;

```

! of routine EDT\$\$RSEQ

```

.TITLE  EDT$WFRESEQ EDT$WFRESEQ - resequence a range of
        lines
.IDENT  \V04-000\
.EXTRN  EDT$$FMT_MSG, EDT$$RD_CURLN
.EXTRN  EDT$$RD_NXTLN, EDT$$L_WK_RESCNT
.EXTRN  EDT$$L_CNOO, EDT$$L_LNO_ZERO
.EXTRN  EDT$$L_LNO_BIG, EDT$$A_CUR_BUF
.EXTRN  EDT$$A_WK_CN, EDT$$G_WK_MODFD
.EXTRN  EDT$$Z_EOB_LN, EDT$_MAXLINNUM
.PSECT  _EDT$CODE,NOWRT, SHR, PIC,2
.ENTRY  EDT$$RSEQ, Save R2,R3,R4,R5,R6,R7,R8,R9,- ; 0664
        R10,R11

```

OFFC 0000

			5E		2C	C2	00002		SUBL2	#44, SP		
			BC		06	28	00005		MOV3	#6, @COUNT, LINE_CNT	0739	
14	AE	04	BC		06	28	00008		MOV3	#6, @COUNT, EDT\$\$L_WK_RESCNT	0740	
00000000G	00	04	BC		6E	D4	00014		CLRL	COPY_FLAG	0741	
			00	14	AE	D1	00016		CMPL	LOW_T, LOW_2	0743	
			00	18	AE	B1	00020		BNEQ	1\$		
			00		03	12	00028		CMPL	HIGH_1, HIGH_2		
					0170	31	0002A		BNEQ	1\$		
					5B	D4	0002D	1\$:	BRW	23\$		0745
0C	AE	08	BC		06	28	0002F		CLRL	EOF		0746
04	AE	08	BC		06	28	00035		MOV3	#6, @START, NEXT_LINE_NUMBER		0747
					59	D4	0003B		MOV3	#6, @START, PREV_NUM		0748
			57	14	AE	D0	0003D		CLRL	LINE_TOO_LARGE		0757
			58	00000000G	00	D0	00041		MOVL	LOW_T, R7		
			56	00000000G	00	D0	00048		MOVL	LOW_2, R8		
			03		5B	E9	0004F	2\$:	MOVL	EDT\$\$A_WK_LN, R6	0769	
					011E	31	00052	3\$:	BLBC	EOF, 4\$	0750	
			58		57	D1	00055	4\$:	BRW	21\$		
					41	12	00058		CMPL	R7, R8		0757
			00	18	AE	B1	0005A		BNEQ	9\$		
					37	12	00062		CMPL	HIGH_1, HIGH_2		
50	00000000G	08	AE	04	01	C1	00064		BNEQ	9\$		
					08	1F	00071		ADDL3	#1, EDT\$\$A_WK_LN, R0	0760	
			04	AE	08	1F	00071		CMPL	4(R0), HIGH_2		
					11	12	00073		BLSSU	5\$		
			50		60	D1	00075		BNEQ	7\$		
					05	1E	00079		CMPL	(R0), LOW_2		
			50		01	CE	0007B	5\$:	BGEQU	6\$		
					09	11	0007E		MNEGL	#1, R0		
					04	12	00080	6\$:	BRB	8\$		
					50	D4	00082		BNEQ	7\$		
			50		03	11	00084		CLRL	R0		
					01	D0	00086	7\$:	BRB	8\$		
					C7	14	00089	8\$:	MOVL	#1, R0		
			00000000G		00	D6	0008B		BGTR	3\$		
					17	12	00091		INCL	FIRST_LWORD	0764	
			00000000G		00	B6	00093		BNEQ	10\$		
					0F	11	00099		INCL	NEXT_WORD		
			50		57	D0	0009B	9\$:	BRB	10\$	0760	
				14	AE	D7	0009E		MOVL	R7, SAVE	0767	
			50	14	AE	D1	000A1		DECL	FIRST_WORD		
					03	1B	000A5		CMPL	FIRST_WORD, SAVE		
				18	AE	B7	000A7		BLEQU	10\$		
01	A6	0C	AE		06	28	000AA	10\$:	DECW	NEXT_WORD		
			00		01	D0	000B0		MOV3	#6, NEXT_LINE_NUMBER, 1(R6)	0769	
			00000000G		00	D0	000B0		MOVL	#1, EDT\$\$G_WK_MODFD	0770	
			00000000G		00	FB	000B7		CALLS	#0, EDT\$\$RD_NXTLN	0771	
			57	14	AE	D0	000BE		MOVL	LOW_1, R7	0773	
			58	00000000G	00	D0	000C2		MOVL	LOW_2, R8		
			58		57	D1	000C9		CMPL	R7, R8		
					1C	12	000CC		BNEQ	11\$		
			00	18	AE	B1	000CE		CMPL	HIGH_1, HIGH_2		
					12	12	000D6		BNEQ	11\$		
			0F		6E	E8	000D8		BLBS	COPY_FLAG, 11\$	0776	
			50	00000000G	00	D0	000DB		MOVL	EDT\$\$A_CUR_BUF, R0	0779	
1C	AE		60		0E	28	000E2		MOV3	#14, (R0), SAV_BUFPOS		
			6E		01	D0	000E7		MOVL	#1, COPY_FLAG	0780	

04	AE	OC	AE	06	28	000EA	11\$:	MOV C3	#6, NEXT_LINE_NUMBER, PREV_NUM	0783
			56	00	00	000F0		MOV L	EDT\$\$A_WR_LN, R6	0785
			50	00	00	000C00G		MOV AB	EDT\$\$Z_EOB_LN, R0	
			50	56	D1	000FE		CMPL	R6, R0	
				6A	13	00101		BEQL	19\$	
	50	OC	AC	06	C1	00103		ADD L3	#6, INC, R0	0788
	51	12	AE	60	A1	00108		ADD W3	(R0), S2 UP, SAVES2	
			50	AE	B0	0010D		MOV W	S2 UP, SAVED	
			52	BC	9E	00111		MOV AB	@INC, R2	
		OC	AE	62	C0	00115		ADD L2	(R2), NEXT_LINE_NUMBER	
		10	AE	A2	D8	00119		ADWC	4(R2), NEXT_LINE_NUMBER	
			51	AE	B1	0011E		CMPL	DEST_UP, SAVES2	
				04	12	00122		BNEQ	12\$	
				5A	D4	00124		CLRL	MAX	
				03	11	00126		BRB	13\$	
			5A	01	D0	00128	12\$:	MOV L	#1, MAX	
		12	AE	50	B0	0012B	13\$:	MOV W	SAVED, DEST_UP	
			50	00	3C	0012F		MOV ZWL	HIGH_2, R0	0790
			50	AE	B1	00136		CMPL	HIGH_1, R0	
				0F	1F	0013A		BLSSU	14\$	
				18	12	0013C		BNEQ	16\$	
			50	00	D0	0013E		MOV L	LOW_2, R0	
			50	AE	D1	00145		CMPL	LOW_1, R0	
				05	1E	00149		BGEQU	15\$	
			50	01	CE	0014B	14\$:	MNEGL	#1, R0	
				09	11	0014E		BRB	17\$	
				04	12	00150	15\$:	BNEQ	16\$	
				50	D4	00152		CLRL	R0	
				03	11	00154		BRB	17\$	
			50	01	D0	00156	16\$:	MOV L	#1, R0	
				04	14	00159	17\$:	BGTR	18\$	
				5A	D5	0015B		TSTL	MAX	
				11	13	0015D		BEQL	20\$	
			59	01	D0	0015F	18\$:	MOV L	#1, LINE_TOO_LARGE	0793
OC	AE	00000000G	00	06	28	00162		MOV C3	#6, EDT\$\$L_LNO_BIG, NEXT_LINE_NUMBER	0794
				03	11	0016B		BRB	20\$	0785
			5B	01	D0	0016D	19\$:	MOV L	#1, EOF	0799
				FEDC	31	00170	20\$:	BRW	2\$	0750
			50	00	D0	00173	21\$:	MOV L	EDT\$\$A_CUR_BUF, R0	0803
	60	1C	AE	0E	28	0017A		MOV C3	#14, SAV_BUFPOS, (R0)	
			00	00	FB	0017F		CALLS	#0, EDT\$\$RD_CURLN	0804
				59	D5	00186		TSTL	LINE_TOO_LARGE	0806
				0F	13	00188		BEQL	22\$	
				00000000G	8F	DD	0018A	PUSHL	#EDT\$ MAXLINNUM	0809
			00	01	FB	00190		CALLS	#1, EDT\$\$FMT_MSG	
				04	11	00197		BRB	23\$	0813
			50	01	D0	00199	22\$:	MOV L	#1, R0	
					04	0019C		RET		
				50	D4	0019D	23\$:	CLRL	R0	0815
				04	0019F			RET		

: Routine Size: 416 bytes, Routine Base: _EDT\$CODE + 0000

: 246 0816 1
: 247 0817 1 !<BLF/PAGE>

EDT\$WFRESEQ
V04-000

EDT\$WFRESEQ - resequence a range of lines
EDT\$SRSEQ - resequence a range of lines

B 15
15-Sep-1984 02:13:45
14-Sep-1984 12:25:43

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

EDT
V04

: 249 0818 1 END
: 250 0819 1
: 251 0820 0 ELUDOM

! of module EDT\$WFRESEQ

PSECT SUMMARY

Name Bytes Attributes
_EDT\$CODE 416 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	53	14	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=LIS\$:WFRESEQ/OPJ=OBJ\$:WFRESEQ MSRC\$:WFRESEQ.BLI/UPDATE=(ENH\$:WFRESEQ)

: Size: 416 code + 0 data bytes
: Run Time: 00:23.2
: Elapsed Time: 00:26.9
: Lines/CPU Min: 2117
: Lexemes/CPU-Min: 12924
: Memory Used: 194 pages
: Compilation Complete

UMSG LIS	WFCOPLIN LIS												
USSTRING LIS												WFSCOPY LIS	
				WFDELLIN LIS	WFGETBKT LIS	WFOPNBUF LIS	WFREABCK LIS	WFREAFWD LIS					
													WFSTRINS LIS
	WFAPPBKT 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												
						WFECOPY LIS							