

File created: 14-Oct-2023 14:53:17 {WMEDLEY}<lispusers>TEDITKEY.;4

edit by: rmk

changes to: (FNS \SEL.LINEDESC)

previous date: 24-Oct-2022 15:25:58 {WMEDLEY}<lispusers>TEDITKEY.;2

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

;;
;; Copyright (c) 1985-1987 by Xerox Corporation.

(RPAQQ TEDITKEYCOMS
[(COMS

;;; This is the Lyric-and-later version of TEditKey
)

```
(COMS ; functions for affecting the selection
(FNS NTHCAR \TEXTOBJ.WINDEX \TK.PREVSCREEN \TK.UNDERLINE.SEL.ON \TK.UNDERLINE.SEL.OFF
\TK.BOLD.SEL.ON \TK.BOLD.SEL.OFF \TK.ITALIC.SEL.ON \TK.ITALIC.SEL.OFF \TK.SMALLERSEL
\TK.LARGERSEL \TK.SUPERSCRIPSEL \TK.SUBSCRIPSEL \TK.DEFAULTSSEL \TK.DEL.WORD.FORWARD
\TK.UCASE.SEL \TK.CAPITALISE.SEL \CAPITALISE \TK.LCASE.SEL)
; functions for affecting the paraloops of the selection
(FNS \TK.CENTER.SEL \TK.CENTER.SEL.REV \TK.NEST \TK.UNNEST))
(COMS ; functions for affecting (and displaying) the caret character looks
(FNS \TK.SHOWCARETLOOKS \TK.BOLD.CARET.ON \TK.BOLD.CARET.OFF \TK.ITALIC.CARET.ON
\TK.ITALIC.CARET.OFF \TK.UNDERLINE.CARET.ON \TK.UNDERLINE.CARET.OFF \TK.SUPERSCRIP.CARET
\TK.SUBSCRIP.CARET \TK.SMALLER.CARET \TK.LARGER.CARET \TK.DEFAULTS.CARET \TK.FONT1 \TK.FONT2
\TK.FONT3 \TK.SETCARETFONT \TK.FONT4 \TK.FONT5 \TK.FONT6 \TK.FONT7 \TK.FONT8)
; the functions which aren't currently used, which toggle the caret
; looks
(FNS \TK.BOLDTOGGLE \TK.ITALICTOGGLE \TK.UNDERLINETOGGLE))
(COMS ; functions dealing with the default looks
(FNS \TK.SETDEFAULTLOOKS))
(COMS ; functions for positioning within a document
(FNS GOTONEXTTTYWINDOW \TK.NEXTLINE \TK.PREVLIN \TK.GOTODOCBEGIN \TK.GOTODOCEND \TK.GOTOLINEBEGIN
\TK.GOTOLINEEND \TK.PREVCHAR \TK.NEXTCHAR \TK.FORWARD.WORD \TK.BACK.WORD \TK.SELECT.ALL))
(COMS ; other utilities
(FNS \TK.FIND \TK.REDISPLAY \TK.DELLINEFORWARD \TK.OPENLINE \TK.DELCHARFORWARD \TK.TRANSPOSECHARS)
)
(COMS ; little selection utilities etc., for building hacks
(FNS \SEL.LIMIT \TK.SETFILEPTR.TO.CARET \SEL.LINEDESC)
(MACROS \SEL.LIMIT.FORWARD \TK.ONOROFF \LINEDESC.LAST.REAL.CHAR))
(COMS ; fns for the key interface itself
(FNS \SHIFTACTION \ACTION TEDITKEY.INSTALL TEDITKEY.DEINSTALL \TK.ACTIONTOCHARCODE \TK.BUILD.MENU
\TK.HELP \TK.SETFONTINLOOKS WRITE.CHARDESC.AUX CHARDESC TEDITKEY.CONFIGURE \TK.ADDKEY
\TK.CHANGEKEY \TK.APPLYPENDING \TK.NTHFONT)
; redefinition of system junk
(FNS METASHIFT))
```

```
;; (\TK.BOLDTOGGLE (##H ##h) 'toggle Bold caret looks') (\TK.ITALICTOGGLE (##i ##I) 'toggle Italic caret looks')
;; (\TK.SUPERSCRIP.CARET (##^) 'Superscript the caret looks') (\TK.SUBSCRIP.CARET (##_) 'Subscript the caret looks')
```

```
(FNS TEDIT.FULL.FIND)
[ VARS \TK.WHITESPACE (TEDIT.INTERRUPTS `((%, (CHARCODE ^G)
ERROR)
(%, (CHARCODE ^C)
HELP]

(CONSTANTS (\TK.WHITESPACE 22))
(INITVARS (TEDITKEY.VERBOSE T)
(TEDITKEY.METAKEY 'TAB)
(TEDITKEY.LOCKTOGGLEKEY NIL)
(TEDITKEY.NESTWIDTH 36)
(\TK.SIZEINCREMENT 2)
(TEDITKEY.OFFSETINCREMENT 3)
(TEDITKEY.FONTS ' (TIMESROMAN HELVETICA GACHA MODERN CLASSIC TERMINAL SYMBOL HIPPO))
(TEDITKEY.FNKEYFLG T))
(MACROS METACODE CONTROLCODE LCMETACODE)
(INITVARS (\TK.SELKEY 'OPEN)
(\TK.PENDING))
[ INITVARS [TEDITKEY.KEYBINDINGS `((\TK.FONT1 (%##1)
%,
(CONCAT "change to font " (\TK.NTHFONT 1)))
(\TK.FONT2 (%##2)
%,
(CONCAT "change to font " (\TK.NTHFONT 2)))
(\TK.FONT3 (%##3)
%,
(CONCAT "change to font " (\TK.NTHFONT 3)))
(\TK.FONT4 (%##4)
%,
(CONCAT "change to font " (\TK.NTHFONT 4)))
```

```

(\TK.FONT5 (###5)
  %,
  (CONCAT "change to font " (\TK.NTHFONT 5)))
(\TK.FONT6 (###6)
  %,
  (CONCAT "change to font " (\TK.NTHFONT 6)))
(\TK.FONT7 (###7)
  %,
  (CONCAT "change to font " (\TK.NTHFONT 7)))
(\TK.FONT8 (###8)
  %,
  (CONCAT "change to font " (\TK.NTHFONT 8)))
NIL
(\TK.DEFAULTS.CARET (###/)
  "restore the default caret looks")
(\TK.SMALLER.CARET (###9)
  "decrease the caret font size")
(\TK.LARGER.CARET (###0)
  "increase the caret font size")
(\TK.SHOWCARETLOOKS (###=)
  "display the current caret looks")
NIL
(\TK.REDISPLAY (###R ###r)
  "Restore the display")
(\TK.HELP (###?)
  "displays the current key bindings")
NIL
(\TK.PREVCHAR (^B ^b)
  "Back one character")
(\TK.NEXTCHAR (^F ^f)
  "Forward one character")
(\TK.FORWARD.WORD (###F ###f)
  "Forward one word")
(\TK.BACK.WORD (###B ###b)
  "Back one word")
(\TK.GOTOLINEBEGIN (^A ^a)
  "go to stArt of line")
(\TK.GOTOLINEEND (^E ^e)
  "go to End of line")
(\TK.PREVLIN ( ^P ^p)
  "go to Previous line")
(\TK.NEXTLINE (^N ^n)
  "go to Next line")
(\TK.GOTODOCBEGIN (###<)
  "start of document")
(\TK.GOTODOCEND (###>)
  "end of document")
(\TK.SELECT.ALL (###S ###s)
  "Select whole document")
NIL
(\TK.DELLINEFORWARD (^K ^k)
  "Kill line")
(\TK.OPENLINE (^O ^o)
  "Open up blank line")
(\TK.DELCHARFORWARD (^D ^d)
  "Delete character forward")
(\TK.DEL.WORD.FORWARD (###D ###d)
  "Delete word forward")
(\TK.TRANSPOSECHARS (^T ^t)
  "Transpose characters")
NIL NIL (\TK.NEST (|##|)
  "indents margins (nest)")
(\TK.UNNEST (|##|)
  "exdents margins (unnest)")
(\TK.CENTER.SEL (###J ###j)
  "alter Justification")
(\TK.UCASE.SEL (###U ###u)
  "Uppercasify selection")
(\TK.CAPITALISE.SEL (###C ###c)
  "Capitalize selection")
(\TK.LCASE.SEL (###L ###l)
  "Lowercasify selection")
(GET.OBJ.FROM.USER (###O ###o)
  "insert Object"]
[TEDITKEY.DLION.KEYACTIONS `((STOP (%, (CHARCODE ^G)
  %,
  (CHARCODE ^C)
  NOLOCKSHIFT))
  (OPEN (%, (CHARCODE 2,1)
  %,
  (CHARCODE 2,41)
  NOLOCKSHIFT))
  (FONT FONTDOWN . FONTUP)
  (KEYBOARD USERMODE1DOWN . USERMODE1UP]
(COMS ;; (TEDITKEY.FNKEYACTIONS (BQUOTE ((CENTER (, (CHARCODE ##^B), (CHARCODE ##^C))) (BOLD (, (CHARCODE
;; ##^D), (CHARCODE ##^E) NOLOCKSHIFT)) (ITALICS (, (CHARCODE ##^F), (CHARCODE ##^G) NOLOCKSHIFT))
;; (UNDERLINE (, (CHARCODE ##^H), (CHARCODE ##^I) NOLOCKSHIFT)) (SUPERSCRIP (, (CHARCODE ##^J),

```

```

;; (CHARCODE ##^K) NOLOCKSHIFT)) (SUBSCRIPT (, (CHARCODE ##^L), (CHARCODE ##^N) NOLOCKSHIFT))
;; (LARGER (, (CHARCODE ##^O), (CHARCODE ##^P) NOLOCKSHIFT)) (DEFAULTS (, (CHARCODE ##^Q),
;; (CHARCODE ##^R) NOLOCKSHIFT)) (BS (, (CHARCODE ^H), (CHARCODE ^D) NOLOCKSHIFT))))
)
[TEDITKEY.DLION.KEYBINDINGS ' (( (\ACTION 'OPEN)
                                \TK.OPENLINE)
                              (( (\ACTION 'HELP)
                                \TK.HELP)
                              (( (\ACTION 'MARGINS)
                                \TK.NEST)
                              (( (\SHIFTACTION 'MARGINS)
                                \TK.UNNEST)
                              (( (\SHIFTACTION 'NEXT)
                                GOTONEXTTTYWINDOW)
[TEDITKEY.FNKEYBINDINGS ' (( (\ACTION 'DEFAULTS)
                              \TK.DEFAULTSSEL)
                              (( (\SHIFTACTION 'DEFAULTS)
                              \TK.SETDEFAULTLOOKS]
(COMS ;; NOT NEEDED (TEDITKEY.DLION.KEYSYNTAX (QUOTE (((\ACTION (QUOTE NEXT)) NEXT) ((\ACTION (QUOTE
;; UNDO)) UNDO) ((\ACTION (QUOTE BS)) CHARDELETE))))
)
[TEDITKEY.DORADO.KEYACTIONS ` ( (BS (% , (CHARCODE ^H)
                                % ,
                                (CHARCODE ^H) ) )
                              (BLANK-BOTTOM (% , (CHARCODE %##^A)
                                % ,
                                (CHARCODE %##^A) ) )
                              (BLANK-TOP FONTDOWN . FONTUP)
                              (BLANK-MIDDLE USERMODEIDOWN . USERMODEIUP)
                              (CENTER (2,101 2,141 NOLOCKSHIFT))
                              (BOLD (2,102 2,142 NOLOCKSHIFT))
                              (ITALICS (2,103 2,143 NOLOCKSHIFT))
                              (UNDERLINE (2,106 2,146 NOLOCKSHIFT))
                              (SUPERSCRIP (2,113 2,153 NOLOCKSHIFT))
                              (SUBSCRIPT (2,114 2,154 NOLOCKSHIFT))
                              (LARGER (2,110 2,150 NOLOCKSHIFT))
                              (DEFAULTS (2,115 2,155 NOLOCKSHIFT]
(TEDITKEY.DORADO.KEYSYNTAX ' (( (CHARCODE %##N)
                                NEXT)
                              ((CHARCODE %##n)
                                NEXT)
                              ((\ACTION 'BLANK-BOTTOM)
                                UNDO)
                              ((\ACTION 'BS)
                                CHARDELETE]
(P (TEDITKEY.INSTALL))
(P (\TK.BUILD.MENU))
(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVERS (ADDVARS (NLAMA)
                                                                    (NLAML)
                                                                    (LAMA METASHIFT]))

```

;;; This is the Lyric-and-later version of TEditKey

;; functions for affecting the selection

(DEFINEQ

(NTHCAR

```

[LAMBDA (LIST N)
  (CAR (NTH LIST N])

```

(* gbn "10-Oct-85 20:54")

(\TEXTOBJ.WINDEX

```

[LAMBDA (TEXTOBJ)

```

(* gbn "10-Oct-85 20:51")

(* * returns the number which is the position in the list textobj%:\window indicating which window had the last selection in it. This number is then an index into line descriptor lists etc.)

```

(bind (CURW _ (fetch SELWINDOW of TEXTOBJ)) for J from 1 as W in (fetch \WINDOW of TEXTOBJ)
until (EQ W CURW) do NIL finally (RETURN J])

```

(\TK.PREVSCREEN

```

[LAMBDA (STREAM TEXTOBJ SEL)

```

(* gbn "10-Oct-85 00:10")

```

(PROG (THIS PREV)
  (TEDIT.NORMALIZECARET STREAM SEL)
  (if [SETQ THIS (CAR (MKLIST (fetch L1 of SEL)
                              then
                                (SETQ PREV (fetch PREVLIN of THIS))
                                (if (ZEROP (fetch CHARLIM of PREV))

```

(* moves the selection up one line)

(* get the selection on the screen so that it has a line descriptor)

(* empty docs have no line descriptors)

(* we need to back format because this is a fake line descriptor)

```

then
  (\BACKFORMAT (CAR (fetch LINES of TEXTOBJ))
                TEXTOBJ
                (fetch SELWINDOW of TEXTOBJ))
  (SETQ PREV (fetch PREVLIN of THIS))

```

(* (SETQ PREV (replace PREVLIN of THIS with

```
(\FORMATLINE TEXTOBJ NIL (ADD1
(fetch CHARLIM of THIS))))]]
(TEDIT.NORMALIZECARET STREAM (TEDIT.SETSEL STREAM (IMIN (IPLUS (fetch CHAR1 of PREV)
(IDIFFERENCE (\SEL.LIMIT SEL)
(fetch CHAR1
of THIS))))
(fetch CHARLIM of PREV))
0])
```

(\TK.UNDERLINE.SEL.ON

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "19-Mar-85 12:01")
(COND
((SHIFTDOWNP 'FONT)
(\TK.FONT4 TEXTSTREAM TEXTOBJ SEL))
((SHIFTDOWNP 'USERMODE1)
(\TK.UNDERLINE.CARET.ON TEXTSTREAM TEXTOBJ SEL))
(T
(* do the real function of \tk.underlineon which happens when neither Keyboard nor font is held)
(TEDIT.LOOKS TEXTSTREAM ' (UNDERLINE ON)
SEL]))
```

(\TK.UNDERLINE.SEL.OFF

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "19-Mar-85 12:01")
(COND
((SHIFTDOWNP 'FONT)
(\TK.FONT4 TEXTSTREAM TEXTOBJ SEL))
((SHIFTDOWNP 'USERMODE1)
(\TK.UNDERLINE.CARET.OFF TEXTSTREAM TEXTOBJ SEL))
(T
(* do the real function of \tk.underlineon which happens when neither Keyboard nor font is held)
(TEDIT.LOOKS TEXTSTREAM ' (UNDERLINE OFF)
SEL]))
```

(\TK.BOLD.SEL.ON

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "19-Mar-85 12:01")
(COND
((SHIFTDOWNP 'FONT)
(\TK.FONT2 TEXTSTREAM TEXTOBJ SEL))
((SHIFTDOWNP 'USERMODE1)
(\TK.BOLD.CARET.ON TEXTSTREAM TEXTOBJ SEL))
(T
(* do the real function of \tk.boldon which happens when neither Keyboard nor font is held)
(* acts on the selection)
(TEDIT.LOOKS TEXTSTREAM ' (WEIGHT BOLD)
SEL]))
```

(\TK.BOLD.SEL.OFF

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL CHARCODE) (* gbn "19-Mar-85 12:02")
(COND
((SHIFTDOWNP 'FONT)
(\TK.FONT2 TEXTSTREAM TEXTOBJ SEL))
((SHIFTDOWNP 'USERMODE1)
(\TK.BOLD.CARET.OFF TEXTSTREAM TEXTOBJ SEL))
(T
(* do the real function of \tk.boldon which happens when neither Keyboard nor font is held)
(* acts on the selection)
(TEDIT.LOOKS TEXTSTREAM ' (WEIGHT MEDIUM)
SEL]))
```

(\TK.ITALIC.SEL.ON

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "19-Mar-85 12:02")
(COND
((SHIFTDOWNP 'FONT)
(\TK.FONT3 TEXTSTREAM TEXTOBJ SEL))
((SHIFTDOWNP 'USERMODE1)
(\TK.ITALIC.CARET.ON TEXTSTREAM TEXTOBJ SEL))
(T (TEDIT.LOOKS TEXTSTREAM ' (SLOPE ITALIC)
SEL]))
```

(\TK.ITALIC.SEL.OFF

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL CHARCODE) (* gbn "19-Mar-85 12:02")
(COND
((SHIFTDOWNP 'FONT)
(\TK.FONT3 TEXTSTREAM TEXTOBJ SEL))
((SHIFTDOWNP 'USERMODE1)
(\TK.ITALIC.CARET.OFF TEXTSTREAM TEXTOBJ SEL))
(T (TEDIT.LOOKS TEXTSTREAM ' (SLOPE REGULAR)
SEL]))
```

(\TK.SMALLERSEL

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 21:47")
(COND
  ((SHIFTDOWNP 'FONT)
   (\TK.FONT7 TEXTSTREAM TEXTOBJ SEL))
  ((SHIFTDOWNP 'USERMODE1)
   (\TK.SMALLER.CARET TEXTSTREAM TEXTOBJ SEL))
(T
  (* do the real function of \TK.SUPERSCRIPSEL which happens when neither Keyboard nor font is held)
  (* acts on the selection)
  (TEDIT.LOOKS TEXTSTREAM (LIST 'SIZEINCREMENT (IMINUS \TK.SIZEINCREMENT))
   SEL]))
```

(\TK.LARGERSEL

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 21:47")
(COND
  ((SHIFTDOWNP 'FONT)
   (\TK.FONT7 TEXTSTREAM TEXTOBJ SEL))
  ((SHIFTDOWNP 'USERMODE1)
   (\TK.LARGER.CARET TEXTSTREAM TEXTOBJ SEL))
(T
  (* do the real function of \TK.LARGER.SEL which happens when neither Keyboard nor font is held)
  (* acts on the selection)
  (TEDIT.LOOKS TEXTSTREAM (LIST 'SIZEINCREMENT \TK.SIZEINCREMENT)
   SEL]))
```

(\TK.SUPERSCRIPSEL

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 21:56")
(COND
  ((SHIFTDOWNP 'FONT)
   (\TK.FONT5 TEXTSTREAM TEXTOBJ SEL))
  ((SHIFTDOWNP 'USERMODE1)
   (\TK.SUPERSCRIP.CARET TEXTSTREAM TEXTOBJ SEL))
(T
  (* do the real function of \TK.SUPERSCRIPSEL which happens when neither Keyboard nor font is held)
  (* acts on the selection)
  (TEDIT.LOOKS TEXTSTREAM (LIST 'OFFSETINCREMENT TEDITKEY.OFFSETINCREMENT)
   SEL]))
```

(\TK.SUBSCRIPSEL

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 21:42")
(COND
  ((SHIFTDOWNP 'FONT)
   (\TK.FONT6 TEXTSTREAM TEXTOBJ SEL))
  ((SHIFTDOWNP 'USERMODE1)
   (\TK.SUBSCRIP.CARET TEXTSTREAM TEXTOBJ SEL))
(T
  (* do the real function of \TK.SUPERSCRIPSEL which happens when neither Keyboard nor font is held)
  (* acts on the selection)
  (TEDIT.LOOKS TEXTSTREAM (LIST 'OFFSETINCREMENT (MINUS TEDITKEY.OFFSETINCREMENT))
   SEL]))
```

(\TK.DEFAULTSSEL

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 21:55")
(* acts on the selection)
(COND
  ((SHIFTDOWNP 'FONT)
   (\TK.FONT8 TEXTSTREAM TEXTOBJ SEL))
  ((SHIFTDOWNP 'USERMODE1)
   (\TK.DEFAULTS.CARET TEXTSTREAM TEXTOBJ SEL))
(T
  (* acts on the selection)
  (PROG ((LOOKS (create CHARLOOKS using TEDIT.DEFAULT.CHARLOOKS)))
   (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
   (TEDIT.LOOKS TEXTSTREAM LOOKS SEL]))
```

(\TK.DEL.WORD.FORWARD

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "20-Mar-85 00:45")
(* * Deletes from here to the end of the first word Refers to the syntax classes of the characters according to the
TEDIT.WORDBOUND.READTABLE)
(PROG (HERE)
(* position the file ptr at the (character after the) caret of the
selection)
(\TK.SETFILEPTR.TO.CARET TEXTSTREAM TEXTOBJ SEL)
(* record this position as the beginning of the word
(to make the beginning of the selection))
(SETQ HERE (ADD1 (GETFILEPTR TEXTSTREAM)))
(* skip the whitespace)
(while [AND (NOT (EOPF TEXTSTREAM))
  (EQ \TK.WHITESPACE (TEDIT.WORDGET (\PEEKBIN TEXTSTREAM T)
  do (BIN TEXTSTREAM))
(* find out what syntax class the first letter of the word has. The end of the word is marked by a change of syntax classes)
```

```
(if (NOT (EOFP TEXTSTREAM))
  then (SETQ CLASS (TEDIT.WORDGET (\PEEKBIN TEXTSTREAM T)))
    (while [AND (NOT (EOFP TEXTSTREAM))
              (EQ CLASS (TEDIT.WORDGET (\PEEKBIN TEXTSTREAM T]
                do (BIN TEXTSTREAM)))
      (TEDIT.SETSEL TEXTSTREAM HERE (ADD1 (IDIFFERENCE (GETFILEPTR TEXTSTREAM)
                                                       HERE)))
      'RIGHT)
  (TEDIT.DELETE TEXTSTREAM)
  (TEDIT.NORMALIZECARET TEXTOBJ])
```

(\TK.UCASE.SEL

```
[LAMBDA (STREAM TEXTOBJ SEL)
```

(* gbn "20-Mar-85 00:46")
 (* uppercasifies the selection)

```
(PROG ((STR (TEDIT.SEL.AS.STRING STREAM SEL))
      (POS (fetch CH# of SEL))
      (LEN (fetch DCH of SEL))
      (POINT (fetch POINT of SEL)))
  (TEDIT.DELETE STREAM SEL)
  (TEDIT.INSERT STREAM (U-CASE STR))
  (TEDIT.SETSEL STREAM POS LEN POINT)
  (TEDIT.NORMALIZECARET TEXTOBJ])
```

(\TK.CAPITALISE.SEL

```
[LAMBDA (STREAM TEXTOBJ SEL)
```

(* gbn "24-Feb-86 16:57")
 (* capitalises the selection)

```
(PROG ((STR (TEDIT.SEL.AS.STRING STREAM SEL))
      (POS (fetch (SELECTION CH#) of SEL))
      (LEN (fetch (SELECTION DCH) of SEL))
      (POINT (fetch (SELECTION POINT) of SEL)))
  (TEDIT.DELETE STREAM SEL)
  (TEDIT.INSERT STREAM (\CAPITALISE STR))
  (TEDIT.SETSEL STREAM POS LEN POINT)
  (TEDIT.NORMALIZECARET TEXTOBJ])
```

(\CAPITALISE

```
[LAMBDA (STR)
```

(* gbn "24-Feb-86 16:56")

(* * capitalises a string)

```
(SELECTQ (NCHARS STR)
  (0 STR)
  (1 (U-CASE STR))
  (CONCAT (U-CASE (NTHCHAR STR 1))
          (L-CASE (SUBSTRING STR 2])))
```

(\TK.LCASE.SEL

```
[LAMBDA (STREAM TEXTOBJ SEL)
```

(* gbn "20-Mar-85 00:46")
 (* uppercasifies the selection)

```
(PROG ((STR (TEDIT.SEL.AS.STRING STREAM SEL))
      (POS (fetch CH# of SEL))
      (LEN (fetch DCH of SEL))
      (POINT (fetch POINT of SEL)))
  (TEDIT.DELETE STREAM SEL)
  (TEDIT.INSERT STREAM (L-CASE STR))
  (TEDIT.SETSEL STREAM POS LEN POINT)
  (TEDIT.NORMALIZECARET TEXTOBJ])
```

)

:: functions for affecting the paraloops of the selection

(DEFINEQ

(\TK.CENTER.SEL

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL)
```

(* gbn "27-Feb-85 18:17")

```
(COND
  ((SHIFTDOWNP 'FONT)
   (\TK.FONT1 TEXTSTREAM TEXTOBJ SEL))
  (T
   (PROG (LOOKS OLDQUAD NEWQUAD NEWQUADS (SAVECH# (fetch CH# of SEL))
         (SAVEDCH (fetch DCH of SEL)))
     (for PARA in (\PARAS.IN.SEL SEL TEXTOBJ)
       do (SETQ LOOKS (TEDIT.GET.PARALOOPS TEXTSTREAM PARA))
          (SETQ OLDQUAD (LISTGET LOOKS 'QUAD))
          [SETQ NEWQUAD (CADR (MEMB OLDQUAD (CONSTANT '(LEFT JUSTIFIED CENTERED LEFT))
                                   (LISTPUT LOOKS 'QUAD NEWQUAD))
          (SETQ PARASEL (TEDIT.SETSEL TEXTSTREAM PARA 1))
          (TEDIT.PARALOOPS TEXTSTREAM LOOKS PARASEL)
          (push NEWQUADS NEWQUAD))
      (TEDIT.SETSEL TEXTSTREAM SAVECH# SAVEDCH)
      (if TEDITKEY.VERBOSE
```

then (TEDIT.PROMPTPRINT TEXTSTREAM (DREVERSE NEWQUADS) T])

(\TK.CENTER.SEL.REV

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "7-Jun-85 21:34")

(COND ((SHIFTDOWNP 'FONT) (\TK.FONT1 TEXTSTREAM TEXTOBJ SEL)) (T (* acts like center.sel but cycles in the opposite direction) (PROG (LOOKS OLDQUAD NEWQUAD NEWQUADS (SAVECH# (fetch CH# of SEL)) (SAVEDCH (fetch DCH of SEL))) (for PARA in (\PARAS.IN.SEL SEL TEXTOBJ) do (SETQ LOOKS (TEDIT.GET.PARALOOKS TEXTSTREAM PARA)) (SETQ OLDQUAD (LISTGET LOOKS 'QUAD)) [SETQ NEWQUAD (CADR (MEMB OLDQUAD (CONSTANT '(LEFT CENTERED JUSTIFIED LEFT)) (LISTPUT LOOKS 'QUAD NEWQUAD)) (SETQ PARASEL (TEDIT.SETSEL TEXTSTREAM PARA 1)) (TEDIT.PARALOOKS TEXTSTREAM LOOKS PARASEL) (push NEWQUADS NEWQUAD)) (TEDIT.SETSEL TEXTSTREAM SAVECH# SAVEDCH) (if TEDITKEY.VERBOSE then (TEDIT.PROMPTPRINT TEXTSTREAM (DREVERSE NEWQUADS) T]))

(\TK.NEST

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Feb-85 18:24")

(PROG (LOOKS (SAVECH# (fetch CH# of SEL)) (SAVEDCH (fetch DCH of SEL))) (for PARA in (\PARAS.IN.SEL SEL TEXTOBJ) do (SETQ LOOKS (TEDIT.GET.PARALOOKS TEXTSTREAM PARA)) (LISTPUT LOOKS 'LEFTMARGIN (IPLUS (LISTGET LOOKS 'LEFTMARGIN) TEDITKEY.NESTWIDTH)) (LISTPUT LOOKS '1STLEFTMARGIN (IPLUS (LISTGET LOOKS '1STLEFTMARGIN) TEDITKEY.NESTWIDTH)) (LISTPUT LOOKS 'RIGHTMARGIN (IMAX 0 (IDIFFERENCE (LISTGET LOOKS 'RIGHTMARGIN) TEDITKEY.NESTWIDTH))) (TEDIT.SETSEL TEXTSTREAM PARA 1) (TEDIT.PARALOOKS TEXTOBJ LOOKS)) (TEDIT.SETSEL TEXTSTREAM SAVECH# SAVEDCH]))

(\TK.UNNEST

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Feb-85 18:26")

(PROG (LOOKS RIGHT (SAVECH# (fetch CH# of SEL)) (SAVEDCH (fetch DCH of SEL))) (for PARA in (\PARAS.IN.SEL SEL TEXTOBJ) do (SETQ LOOKS (TEDIT.GET.PARALOOKS TEXTSTREAM PARA)) (LISTPUT LOOKS 'LEFTMARGIN (IMAX (IDIFFERENCE (LISTGET LOOKS 'LEFTMARGIN) TEDITKEY.NESTWIDTH) 0)) (LISTPUT LOOKS '1STLEFTMARGIN (IMAX (IDIFFERENCE (LISTGET LOOKS '1STLEFTMARGIN) TEDITKEY.NESTWIDTH) 0)) (SETQ RIGHT (LISTGET LOOKS 'RIGHTMARGIN)) (if (NOT (ZEROP RIGHT)) then (LISTPUT LOOKS 'RIGHTMARGIN (IPLUS (LISTGET LOOKS 'RIGHTMARGIN) TEDITKEY.NESTWIDTH))) (TEDIT.SETSEL TEXTSTREAM PARA 1) (TEDIT.PARALOOKS TEXTOBJ LOOKS)) (TEDIT.SETSEL TEXTSTREAM SAVECH# SAVEDCH]))

)

:: functions for affecting (and displaying) the caret character looks

(DEFINEQ

(\TK.SHOWCARETLOOKS

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "30-Jan-85 16:06")

(* comment) (PROG ((LOOKS (fetch CARETLOOKS of TEXTOBJ)) (TEDIT.PROMPTPRINT TEXTSTREAM (CONCAT (\TK.DESCRIBEFONT (fetch CLFONT of LOOKS)) (if (AND (fetch CLOFFSET of LOOKS) (NEQ (fetch CLOFFSET of LOOKS) 0)) then (CONCAT " offset " (fetch CLOFFSET of LOOKS)) else ""))

```

      (if (fetch CLOLINE of LOOKS)
          then " overlined"
          else "")
      (if (fetch CLULINE of LOOKS)
          then " underlined"
          else "")

```

```

      T)
    (RETURN))

```

(\TK.BOLD.CARET.ON

```

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 20:54")
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
    (replace CLBOLD of LOOKS with T)
    (if (\TK.SETFONTINLOOKS TEXTSTREAM LOOKS)
        then (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
        (if TEDITKEY.VERBOSE
            then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
        (RETURN)
      else (RETURN))

```

(\TK.BOLD.CARET.OFF

```

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 20:54")
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
    (replace CLBOLD of LOOKS with NIL)
    (if (\TK.SETFONTINLOOKS TEXTSTREAM LOOKS)
        then (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
        (if TEDITKEY.VERBOSE
            then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
        (RETURN)
      else (RETURN))

```

(\TK.ITALIC.CARET.ON

```

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 22:20")
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
    (replace CLITAL of LOOKS with T)
    (if (\TK.SETFONTINLOOKS TEXTSTREAM LOOKS)
        then (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
        (if TEDITKEY.VERBOSE
            then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
        (RETURN)
      else (RETURN))

```

(\TK.ITALIC.CARET.OFF

```

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 22:19")
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
    (replace CLITAL of LOOKS with NIL)
    (if (\TK.SETFONTINLOOKS TEXTSTREAM LOOKS)
        then (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
        (if TEDITKEY.VERBOSE
            then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
        (RETURN)
      else (RETURN))

```

(\TK.UNDERLINE.CARET.ON

```

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 17:59")
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
    (replace CLULINE of LOOKS with T)
    (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
    (if TEDITKEY.VERBOSE
        then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
    (RETURN))

```

(\TK.UNDERLINE.CARET.OFF

```

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 18:01")
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
    (replace CLULINE of LOOKS with NIL)
    (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
    (if TEDITKEY.VERBOSE
        then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
    (RETURN))

```

(\TK.SUPERSCRIP.T.CARET

```

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 22:25")
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
    (if (fetch CLOFFSET of LOOKS)
        then (add (fetch CLOFFSET of LOOKS)
            TEDITKEY.OFFSETINCREMENT)
        else (replace CLOFFSET of LOOKS with TEDITKEY.OFFSETINCREMENT))
    (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
    (if TEDITKEY.VERBOSE

```

then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL)
(RETURN)]

(\TK.SUBSCRIPT.CARET

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 22:26")
(PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
(if (fetch CLOFFSET of LOOKS)
then (add (fetch CLOFFSET of LOOKS)
(IMINUS TEDITKEY.OFFSETINCREMENT))
else (replace CLOFFSET of LOOKS with (IMINUS TEDITKEY.OFFSETINCREMENT)))
(TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
(if TEDITKEY.VERBOSE
then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
(RETURN)])

(\TK.SMALLER.CARET

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 21:45")
(PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
(replace CLSIZE of LOOKS with (IMAX 4 (IDIFFERENCE (fetch CLSIZE of LOOKS)
2)))
(if (NOT (\TK.SETFONTINLOOKS TEXTSTREAM LOOKS))
then (RETURN))
(TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
(if TEDITKEY.VERBOSE
then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
(RETURN)])

(\TK.LARGER.CARET

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 22:37")
(PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
(replace CLSIZE of LOOKS with (IPLUS \TK.SIZEINCREMENT (fetch CLSIZE of LOOKS)))
(if (NOT (\TK.SETFONTINLOOKS TEXTSTREAM LOOKS))
then (RETURN))
(TEDIT.CARETLOOKS TEXTSTREAM LOOKS)
(if TEDITKEY.VERBOSE
then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
(RETURN)])

(\TK.DEFAULTS.CARET

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "16-Feb-85 21:54")
(PROGN (TEDIT.CARETLOOKS TEXTSTREAM (create CHARLOOKS using TEDIT.DEFAULT.CHARLOOKS))
(if TEDITKEY.VERBOSE
then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL]))

(\TK.FONT1

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Jan-85 17:39")
(\TK.SETCARETFONT TEXTOBJ (CAR (NTH TEDITKEY.FONTS 1]))

(\TK.FONT2

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Jan-85 17:43")
(\TK.SETCARETFONT TEXTOBJ (CAR (NTH TEDITKEY.FONTS 2]))

(\TK.FONT3

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Jan-85 17:40")
(\TK.SETCARETFONT TEXTOBJ (CAR (NTH TEDITKEY.FONTS 3]))

(\TK.SETCARETFONT

[LAMBDA (TEXTOBJ FONTNAME) (* gbn "19-Mar-85 12:02")

(* temporary hack. If this function is called when the keyboard shift is down, then it refers to the caret looks, otherwise the selection)

(if (SHIFTDOWNP 'USERMODEL)
then [PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
(replace CLNAME of LOOKS with FONTNAME)
(if (\TK.SETFONTINLOOKS TEXTOBJ LOOKS)
then (* we found the font, install it as the caret font and tell the user)
(if TEDITKEY.VERBOSE
then (TEDIT.PROMPTPRINT TEXTOBJ FONTNAME T))
(RETURN (TEDIT.CARETLOOKS TEXTSTREAM LOOKS])
else (TEDIT.LOOKS TEXTSTREAM (LIST 'FAMILY FONTNAME)
SEL)])

(\TK.FONT4

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Jan-85 17:40")
(\TK.SETCARETFONT TEXTOBJ (CAR (NTH TEDITKEY.FONTS 4]))

```
(\TK.FONT5
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Jan-85 17:41")
  (\TK.SETCARETFONT TEXTOBJ (CAR (NTH TEDITKEY.FONTS 5]))
```

```
(\TK.FONT6
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Jan-85 17:41")
  (\TK.SETCARETFONT TEXTOBJ (CAR (NTH TEDITKEY.FONTS 6]))
```

```
(\TK.FONT7
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Jan-85 17:42")
  (\TK.SETCARETFONT TEXTOBJ (CAR (NTH TEDITKEY.FONTS 7]))
```

```
(\TK.FONT8
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "27-Jan-85 17:43")
  (\TK.SETCARETFONT TEXTOBJ (CAR (NTH TEDITKEY.FONTS 8]))
```

)

:: the functions which aren't currently used, which toggle the caret looks

(DEFINEQ

```
(\TK.BOLDTOGGLE (* gbn "16-Feb-85 20:54")
[LAMBDA (TEXTSTREAM TEXTOBJ SEL)
  (** toggles boldness in the caret looks)

  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
(replace CLBOLD of LOOKS with (NOT (fetch CLBOLD of LOOKS)))
(if (\TK.SETFONTINLOOKS TEXTSTREAM LOOKS)
  then (if TEDITKEY.VERBOSE
    then (TEDIT.PROMPTPRINT TEXTSTREAM (CONCAT "bold: " (\TK.ONOROFF (fetch CLBOLD of LOOKS)))
      T))
    (RETURN (TEDIT.CARETLOOKS TEXTSTREAM LOOKS))
  else (RETURN)])
```

```
(\TK.ITALICTOGGLE (* gbn "27-Jan-85 19:23")
[LAMBDA (TEXTSTREAM TEXTOBJ SEL)
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
(replace CLITAL of LOOKS with (NOT (fetch CLITAL of LOOKS)))
(if (\TK.SETFONTINLOOKS TEXTSTREAM LOOKS)
  then (if TEDITKEY.VERBOSE
    then (TEDIT.PROMPTPRINT TEXTSTREAM (CONCAT "italic: " (\TK.ONOROFF (fetch CLITAL
of LOOKS)))
      T))
    (RETURN (TEDIT.CARETLOOKS TEXTSTREAM LOOKS))
  else (RETURN)])
```

```
(\TK.UNDERLINETOGGLE (* gbn "27-Jan-85 19:23")
[LAMBDA (TEXTSTREAM TEXTOBJ SEL)
  (PROG [(LOOKS (create CHARLOOKS using (fetch CARETLOOKS of TEXTOBJ)
(replace CLULINE of LOOKS with (NOT (fetch CLULINE of LOOKS)))
(if TEDITKEY.VERBOSE
  then (TEDIT.PROMPTPRINT TEXTSTREAM (CONCAT "underline: " (\TK.ONOROFF (fetch CLULINE of LOOKS)))
    T))
  (RETURN (TEDIT.CARETLOOKS TEXTSTREAM LOOKS)])
```

)

:: functions dealing with the default looks

(DEFINEQ

```
(\TK.SETDEFAULTLOOKS (* gbn "16-Feb-85 23:00")
[LAMBDA (TEXTSTREAM TEXTOBJ SEL)
  (** sets TEDIT.DEFAULT.CHARLOOKS to have the looks of the current selection)

  (PROG NIL
    (SETQ TEDIT.DEFAULT.CHARLOOKS (COPY (fetch CARETLOOKS of TEXTOBJ)))
    (if TEDITKEY.VERBOSE
      then (\TK.SHOWCARETLOOKS TEXTSTREAM TEXTOBJ SEL))
    (RETURN))
```

)

:: functions for positioning within a document

(DEFINEQ

```
(GOTONEXTTTYWINDOW
```

[LAMBDA NIL (* gbn "7-May-85 16:19")

(* * puts the tty in the next appropriate process in the chain)

```

(PROG ((CURRENT (TTY.PROCESS)))
      (SETQ CANDIDATES (LIST NIL))
      [MAP.PROCESSES (FUNCTION (LAMBDA (PROC)
                              (PROG (W)
                                     (if (AND (SETQ W (PROCESSPROP PROC 'WINDOW))
                                             (OPENWP W)
                                             (WINDOWPROP W 'PROCESS))
                                         then (NCONC1 CANDIDATES PROC]
                              (SETQ NEW (CDR (MEMBER CURRENT CANDIDATES)))
                              (SETQ NEW (if NEW
                                             then (CAR NEW)
                                             else (CADR CANDIDATES))))
      (TTY.PROCESS NEW)
      (FLASHWINDOW (PROCESSPROP NEW 'WINDOW)
                    1 1 GRAYSHADE)

```

(* for (PROC CURRENT) repeatwhile
 (NEQ PROC CURRENT) do (SETQ W
 (PROCESSPROP (SETQ PROC
 (fetch NEXTPROCHANDLE of PROC))
 (QUOTE WINDOW))) (PRINTOUT T
 (PROCESSPROP PROC (QUOTE NAME)))
 (if (AND W (OPENWP W) (WINDOWPROP W
 (QUOTE PROCESS))) then (* this window would probably be
 willing to take the tty if clicked in, so give the process the tty)
 (TTY.PROCESS PROC) (FLASHWINDOW W 1 NIL
 GRAYSHADE) (RETURN)))

])

(\TK.NEXTLINE

[LAMBDA (STREAM TEXTOBJ SEL)

```

(PROG (THIS NEXT)
      (TEDIT.NORMALIZECARET STREAM SEL)
      (SETQ THIS (\SEL.LINEDESC SEL))
      (if THIS
          then
            (SETQ NEXT (fetch NEXTLINE of THIS))
            (if (NOT NEXT)
                then
                  (SETQ NEXT (replace NEXTLINE of THIS with (\FORMATLINE TEXTOBJ NIL
                                                                    (ADD1 (fetch CHARLIM of THIS)
                                                                    0
                                                                    'LEFT)))
                (if NEXT
                    then

```

(* gbn "10-Oct-85 22:47")
 (* moves the selection down one line)
 (* get the selection on the screen so that it has a line descriptor)

(* an empty doc has no line descriptors, even after normalizing)

(* there isn't already a descriptor for this line)

(* if there are no more characters, then there still may not be a descriptor when we call \formatline)

```

(TEDIT.NORMALIZECARET STREAM (TEDIT.SETSEL STREAM
                              (IMIN (IPLUS (fetch CHAR1 of NEXT)
                                             (IDIFFERENCE (\SEL.LIMIT SEL)
                                                           (fetch CHAR1 of THIS)))
                              (ADD1 (\LINEDESC.LAST.REAL.CHAR NEXT)))
                              0
                              'LEFT))

```

(\TK.PREVLIN

[LAMBDA (STREAM TEXTOBJ SEL)

```

(PROG (THIS PREV)
      (TEDIT.NORMALIZECARET STREAM SEL)
      (if (SETQ THIS (\SEL.LINEDESC SEL))
          then
            (SETQ PREV (fetch PREVLIN of THIS))
            (if (ZEROP (fetch CHARLIM of PREV))
                then
                  [\BACKFORMAT (NTHCAR (fetch LINES of TEXTOBJ)
                                     (\TEXTOBJ.WINDEX TEXTOBJ))
                  TEXTOBJ
                  (fetch PTOP of (DSPCLIPPINGREGION NIL (fetch SELWINDOW of TEXTOBJ]
                  (SETQ PREV (fetch PREVLIN of THIS))

```

(* gbn "10-Oct-85 22:15")
 (* moves the selection up one line)
 (* get the selection on the screen so that it has a line descriptor)

(* empty docs have no line descriptors)

(* we need to back format because this is a fake line descriptor)

```

(TEDIT.NORMALIZECARET STREAM (TEDIT.SETSEL STREAM (IMIN (IPLUS (fetch CHAR1 of PREV)
                                                                  (IDIFFERENCE (\SEL.LIMIT SEL)
                                                            (fetch CHAR1
                                                            of THIS)))
                                                                  (fetch CHARLIM of PREV))

```

0])

(\TK.GOTODOCBEGIN

[LAMBDA (STREAM TEXTOBJ SEL)

(* gbn "13-Dec-84 11:24")
 (* positions at the beginning of a document)

(TEDIT.SETSEL STREAM 0 0)
(TEDIT.NORMALIZECARET TEXTOBJ))

(\TK.GOTODOCEND

[LAMBDA (STREAM TEXTOBJ SEL)

(* gbn "7-Jun-85 16:32")
(* positions at the end of a document)

(TEDIT.SETSEL STREAM (ADD1 (fetch TEXTLEN of TEXTOBJ))
0
'LEFT)
(TEDIT.NORMALIZECARET STREAM])

(\TK.GOTOLINEBEGIN

[LAMBDA (TEXTSTREAM TEXTOBJ SEL)

(* gbn "11-Mar-85 15:04")

(* * positions the cursor at the beginning of line)

(PROG (CH)
(TEDIT.NORMALIZECARET TEXTOBJ SEL)
(SETQ CH (fetch CHAR1 of (\SEL.LINEDESC SEL))))

(* (if (fetch CR\END of (fetch L1 of SEL)) then
* there is a CR at the end of this line, we want to position before

it) (SETQ CH (SUB1 CH))))
(TEDIT.SETSEL TEXTSTREAM CH 0 'LEFT])

(\TK.GOTOLINEEND

[LAMBDA (TEXTSTREAM TEXTOBJ SEL)

(* gbn "7-Jun-85 15:47")

(* * positions the cursor at the end of line)

(PROG ((POINT 'RIGHT)
LN)
(TEDIT.NORMALIZECARET TEXTOBJ SEL)
(if (SETQ LN (\SEL.LINEDESC SEL))
then
(SETQ CH (fetch CHARLIM of LN))
(if (fetch CR\END of LN)
then

(* empty docs have no linedescriptors)

(* there is not a CR at the end of this line, we want to position to the right of the last char)

(SETQ POINT 'LEFT))
(TEDIT.SETSEL TEXTSTREAM CH 1 POINT])

(\TK.PREVCHAR

[LAMBDA (STREAM TEXTOBJ SEL)

(* gbn "19-Mar-85 12:16")
(* moves the selection back one char)

(PROG NIL
(TEDIT.SETSEL STREAM (IMAX 0 (SUB1 (\SEL.LIMIT.FORWARD SEL)))
0
'LEFT)

(* I don't think this should be necessary, but there are cases where the caret is not normalised)

(TEDIT.NORMALIZECARET TEXTOBJ SEL])

(\TK.NEXTCHAR

[LAMBDA (STREAM TEXTOBJ SEL)

(* gbn "20-Mar-85 00:48")
(* moves the selection back one char)
(* Note%: addition. does *not* distribute with Min Do not
pessimize this!)

(PROG NIL
(TEDIT.SETSEL STREAM (IMIN (ADD1 (fetch TEXTLEN of TEXTOBJ))
(ADD1 (\SEL.LIMIT.FORWARD SEL)))
0
'LEFT)
(TEDIT.NORMALIZECARET TEXTOBJ])

(\TK.FORWARD.WORD

[LAMBDA (TEXTSTREAM TEXTOBJ SEL)

(* gbn "20-Mar-85 00:48")

(* * moves the caret one word forward. Refers to the syntax classes of the characters according to the
TEDIT.WORDBOUND.READTABLE)

(PROG (HERE)
(\TK.SETFILEPTR.TO.CARET TEXTSTREAM TEXTOBJ SEL)
(while [AND (NOT (EOPF TEXTSTREAM))
(EQ \TK.WHITESPACE (TEDIT.WORDGET (\PEEKBIN TEXTSTREAM T))
do (BIN TEXTSTREAM))
(SETQ HERE (ADD1 (GETFILEPTR TEXTSTREAM))))

(* position the file ptr at the (character after the) caret of the
selection)

(* skip the whitespace)

(* record this position as the beginning of the word
(to make the beginning of the selection))

(* find out what syntax class the first letter of the word has. The end of the word is marked by a change of syntax classes)

```
(if (NOT (EOPF TEXTSTREAM))
  then (SETQ CLASS (TEDIT.WORDGET (\PEEKBIN TEXTSTREAM T)))
        (while [AND (NOT (EOPF TEXTSTREAM))
                    (EQ CLASS (TEDIT.WORDGET (\PEEKBIN TEXTSTREAM T]
                    do (BIN TEXTSTREAM))
          (TEDIT.SETSEL TEXTSTREAM HERE (ADD1 (IDIFFERENCE (GETFILEPTR TEXTSTREAM)
                                                            HERE)))
          'RIGHT)
      (TEDIT.NORMALIZECARET TEXTOBJ])
```

(\TK.BACK.WORD

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "20-Mar-85 00:49")
```

(* * moves the caret one word back Refers to the syntax classes of the characters according to the TEDIT.WORDBOUND.READTABLE)

```
(PROG (HERE) (* position the file ptr at the (character after the) caret of the selection)
  (\TK.SETFILEPTR.TO.CARET TEXTSTREAM TEXTOBJ SEL) (* skip the whitespace)
  (while [AND (NOT (ZEROP (GETFILEPTR TEXTSTREAM)))
            (EQ \TK.WHITESPACE (TEDIT.WORDGET (\BACKPEEKBIN TEXTSTREAM T]
            do (\BACKBIN TEXTSTREAM)) (* record this position as the beginning of the word (to make the beginning of the selection))
          (SETQ HERE (GETFILEPTR TEXTSTREAM))
```

(* find out what syntax class the last letter of the word has. The end of the word is marked by a change of syntax classes)

```
(if (NOT (ZEROP (GETFILEPTR TEXTSTREAM)))
  then (SETQ CLASS (TEDIT.WORDGET (\BACKPEEKBIN TEXTSTREAM T)))
        (while [AND (NOT (ZEROP (GETFILEPTR TEXTSTREAM)))
                    (EQ CLASS (TEDIT.WORDGET (\BACKPEEKBIN TEXTSTREAM T]
                    do (\BACKBIN TEXTSTREAM))
          (TEDIT.SETSEL TEXTSTREAM (ADD1 (GETFILEPTR TEXTSTREAM))
            (IDIFFERENCE HERE (GETFILEPTR TEXTSTREAM))
            'LEFT)
          (TEDIT.NORMALIZECARET TEXTOBJ])
```

(\TK.SELECT.ALL

```
[LAMBDA (STREAM TEXTOBJ SEL) (* gbn "24-Feb-86 17:11")
  (* positions at the end of a document)
```

```
(TEDIT.SETSEL STREAM 0 (ADD1 (fetch TEXTLEN of TEXTOBJ)
  'LEFT])
```

)

:: other utilities

(DEFINEQ

(\TK.FIND

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "10-Dec-84 12:38")
  (* just calls the normal tedit.find starting at the right of the current selection)
```

```
(TEDIT.FULL.FIND TEXTSTREAM])
```

(\TK.REDISPLAY

```
[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "12-Mar-85 14:27")
```

(* * simply redisplay the window in question.)

```
(\TEDIT.REPAINTFN (CAR (MKATOM (fetch \WINDOW of TEXTOBJ]))
```

(\TK.DELLINEFORWARD

```
[LAMBDA (STREAM TEXTOBJ SEL) (* gbn "13-Dec-84 11:56")
  (* deletes from the caret to the end of this line)
```

```
(PROG (HERE DESC)
  (TEDIT.NORMALIZECARET TEXTOBJ)
  (SETQ HERE (\SEL.LIMIT.FORWARD SEL))
  (SETQ DESC (\SEL.LINEDESC SEL))
  (SETQ SEL (TEDIT.SETSEL STREAM HERE (IDIFFERENCE (fetch CHARLIM of DESC)
                                                    HERE)))
  (TEDIT.DELETE STREAM SEL])
```

(\TK.OPENLINE

```
[LAMBDA (STREAM TEXTOBJ SEL) (* gbn "30-Jan-85 18:36")
```

```
(TEDIT.INSERT STREAM (CONSTANT (CHARCODE EOL)))
(\TK.PREVCHAR STREAM TEXTOBJ SEL])
```

(\TK.DELCHARFORWARD

```
[LAMBDA (STREAM TEXTOBJ SEL) (* gbn "20-Mar-85 00:50")
```

(* * deletes one character forward from the caret)

```
(PROG (HERE)
  (SETQ SEL (TEDIT.SETSEL STREAM (\SEL.LIMIT.FORWARD SEL)
    1))
  (TEDIT.DELETE STREAM SEL)
  (TEDIT.NORMALIZECARET TEXTOBJ])
```

(\TK.TRANSPOSECHARS

[LAMBDA (STREAM TEXTOBJ SEL) (* gbn "20-Mar-85 00:51")

(* * transposes the two characters on either side of the point, unless it is the end of a line, in which case it transposes the two characters before the point)

```
(PROG ((KEEPCHARPOS (\SEL.LIMIT.FORWARD SEL))
  KEEPCHAR LINEDESC)
  (TEDIT.NORMALIZECARET TEXTOBJ SEL) (* get the line that the point of the selection is on)
  (SETQ LINEDESC (\SEL.LINEDESC SEL))
  (if (ILESSP (\LINEDESC.LAST.REAL.CHAR LINEDESC)
    KEEPCHARPOS)
    then
    (* the point is after the last real char on this line, so transpose the two before the point.)
    (add KEEPCHARPOS -1))
  (SETQ KEEPCHAR (TEDIT.SEL.AS.STRING STREAM (TEDIT.SETSEL STREAM KEEPCHARPOS 1)))
  (if (AND (IGREATERP KEEPCHARPOS 1)
    (IGEQ (fetch TEXTLEN of TEXTOBJ)
      KEEPCHARPOS))
    then (TEDIT.DELETE STREAM)
    (TEDIT.INSERT STREAM KEEPCHAR (SUB1 KEEPCHARPOS))
    (TEDIT.SETSEL STREAM KEEPCHARPOS 1 'RIGHT))
  (TEDIT.NORMALIZECARET TEXTOBJ SEL])
```

)
;; little selection utilities etc., for building hacks

(DEFINEQ

(\SEL.LIMIT

[LAMBDA (SEL) (* gbn " 8-Mar-85 12:58")

(* returns the character that delimits this selection. The first char if the point is left else the last)

```
(if (EQ (fetch POINT of SEL)
  'LEFT)
  then (fetch CH# of SEL)
  else (SUB1 (fetch CHLIM of SEL]))
```

(\TK.SETFILEPTR.TO.CARET

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "23-Feb-85 15:24")

(* * makes sure that the fileptr is positioned at character on the right of the CARET of the selection) (* NOTE THAT FILEPTR's are one less than the corresponding

```
char# in a sel)
(SETFILEPTR TEXTSTREAM (SUB1 (\SEL.LIMIT.FORWARD SEL]))
```

(\SEL.LINEDESC

[LAMBDA (SEL) ; Edited 14-Oct-2023 14:53 by rmk
; Edited 24-Oct-2022 15:24 by rmk
(* gbn "10-Oct-85 20:57")

;;; Returns the line descriptor of the point of the selection in the last selected window

```
(NTHCAR (if (EQ (fetch (SELECTION POINT) of SEL)
  'LEFT)
  then (fetch (SELECTION L1) of SEL)
  else (fetch (SELECTION LN) of SEL))
  (\TEXTOBJ.WINDEX (fetch (SELECTION SELTEXTOBJ) of SEL]))
```

)
(DECLARE%: EVAL@COMPILE

(PUTPROPS \SEL.LIMIT.FORWARD MACRO [LAMBDA (SEL) (* gbn "13-Dec-84 11:43")
(* returns the character in front of the caret
(ch# for left and chlim for right))

```
(if (EQ (fetch POINT of SEL)
  'LEFT)
  then (fetch CH# of SEL)
  else (fetch CHLIM of SEL]))
```

```
(PUTPROPS \TK.ONOROFF MACRO [LAMBDA (FLG)
  (if FLG
    then "on"
    else "off"])
```

```
(PUTPROPS \LINEDESC.LAST.REAL.CHAR MACRO [LAMBDA (LINEDESC)
  (if (fetch CR\END of LINEDESC)
    then (* there is a CR at the end so the last real char CHLIM-1)
      (SUB1 (fetch CHARLIM of LINEDESC))
    else (fetch CHARLIM of LINEDESC])
)
```

;; fns for the key interface itself

(DEFINEQ

(\SHIFTACTION

[LAMBDA (KEYNAME)

(* gbn "16-Feb-85 16:55")

(* returns the character code generated by this keyname when typed shifted)

(CADAR (KEYACTION KEYNAME])

(\ACTION

[LAMBDA (KEYNAME)

(* gbn "16-Feb-85 16:54")

(* RETURNS THE CHARACTER CODE GENERATED BY THIS

KEYNAME WHEN TYPED UNSHIFTED)

(CAAR (KEYACTION KEYNAME])

(TEDITKEY.INSTALL

[LAMBDA (READTABLE)

(* gbn " 1-Apr-86 22:36")

(* * installs the TEDITKEYBINDINGS on the readtable)

(PROG [(READTABLE (OR READTABLE TEDIT.READTABLE))

INTERRUPT

(FNKEYITEM '(Function% Keys 'BUILDFNKEYS "Bring up the DLion fn keys window"]

(* I think that in Koto, all this is done by the system. The times, they are a-changin'!

(PROGN (* Tell everyone who cares to let ^h be the backspace character)

(if (SETQ INTERRUPT (GETINTERRUPT (CHARCODE ^H))) then

(printout T "Interrupt on ^H disabled") (SETINTERRUPT (CHARCODE ^H)

(QUOTE NIL))) (SETSYNTAX 8 (QUOTE CHARDELETE) \PROMPTFORWORDTTBL)

(SETSYNTAX 8 (QUOTE CHARDELETE) ASKUSERTTBL) (SETSYNTAX 8

(QUOTE CHARDELETE) \ORIGTERMTABLE) (SETSYNTAX 8

(QUOTE CHARDELETE) \PRIMTERMTABLE) (SETSYNTAX 8

(QUOTE CHARDELETE) DEDITTTBL) (SETINTERRUPT (CHARCODE ^G)

(QUOTE HELP))))

(METASHIFT T)

(* TeditKey redefines METASHIFT to operate on

TEDITKEY.METAKEY instead of the swat

(bottom-blank) key)

(* install the functions on the main keyboard, that is, not the extra dlion keys)

[for TRIPLE in TEDITKEY.KEYBINDINGS

do (COND

(TRIPLE

(for KEY in (CADR TRIPLE) do (APPLY* 'TEDIT.SETFUNCTION (EVAL `(CHARCODE %, KEY))

(CAR TRIPLE)

READTABLE]

(* NILs in the list are for formatting the menu)

(MODIFY.KEYACTIONS TEDITKEY.FNKEYACTIONS))

(* install the nextttywindow hack)

(* INTERRUPTCHAR (\SHIFTACTION

(QUOTE NEXT)) (QUOTE (GOTONEXTTTYWINDOW)))

(* So that non-tedits know about the game)

)

(SELECTQ (MACHINETYPE)

(DANDELION [if TEDITKEY.LOCKTOGGLEKEY

then (KEYACTION TEDITKEY.LOCKTOGGLEKEY '(LOCKTOGGLE)

(if (AND (BOUNDP 'DLIONFNKEYS)

(OPENWP DLIONFNKEYS))

then

(CLOSEW DLIONFNKEYS))

(* this machine has real fn keys so close the fake ones)

(* adjust so that the dlion extra keys return meta control codes)

(MODIFY.KEYACTIONS TEDITKEY.DLION.KEYACTIONS)

(for PAIR in TEDITKEY.DLION.KEYBINDINGS do (TEDIT.SETFUNCTION (EVAL (CAR PAIR))

(CADR PAIR)

READTABLE))

(* hang functions off the dlion extra keys

(e.g. italics, bold))

(for PAIR in TEDITKEY.FNKEYBINDINGS do (TEDIT.SETFUNCTION (EVAL (CAR PAIR))

(CADR PAIR)

READTABLE))

(* set next to do next, undo to do undo etc)

(* unnecessary in KOTO (for PAIR in

TEDITKEY.DLION.KEYSYNTAX do

```

(TEDIT.SETSYNTAX (EVAL (CAR PAIR))
(CADR PAIR) READTABLE))
(* remove the menu item that may have already been installed)
(* you can remove non-existent items with impunity)
(TEDIT.REMOVE.MENUITEM TEDIT.DEFAULT.MENU FNKEYITEM)
(PROGN
  (INTERRUPTCHAR (\SHIFACTION 'NEXT)
    ' (GOTONEXTTTYWINDOW)) (* So that non-tedits know about the game)
  ))
(PROGN (MODIFY.KEYACTIONS TEDITKEY.DORADO.KEYACTIONS)
  (for PAIR in TEDITKEY.DORADO.KEYSYNTAX do (TEDIT.SETSYNTAX (EVAL (CAR PAIR))
    (CADR PAIR)
    READTABLE))
  (for PAIR in TEDITKEY.FNKEYBINDINGS do (TEDIT.SETFUNCTION (EVAL (CAR PAIR))
    (CADR PAIR)
    READTABLE))
  (TEDIT.ADD.MENUITEM TEDIT.DEFAULT.MENU FNKEYITEM))
(SELECTQ (MACHINETYPE)
  ((DORADO DOLPHIN)
    (FILESLOAD (SYSLOAD FROM VALUEOF LISPUSERSDIRECTORIES)
      DLIONFNKEYS)
  (COND
    ([AND TEDITKEY.FNKEYFLG (NOT (AND (BOUNDP 'DLIONFNKEYS)
      (OPENWP DLIONFNKEYS))
    (* if he has the flag set to do so, then check if there is a fnkey window up yet, and build one if there isn't)
    (BUILDFNKEYS)))
  (PROGN NIL))
(* install the forms necessary to re-establish the correct bindings on a new machine if this is sysout'ed)
(* if (NOT (ASSOC (QUOTE TEDITKEY.INSTALL)
  AFTERMAKESYSFORMS)) then
  (push AFTERMAKESYSFORMS
    (QUOTE (TEDITKEY.INSTALL))))
(COND
  ((NOT (ASSOC 'TEDITKEY.INSTALL AFTERSYSOUTFORMS))
    (push AFTERSYSOUTFORMS ' (TEDITKEY.INSTALL]
  (RETURN (CONCAT TEDITKEY.METAKEY "'s action is now Meta. TEditKey actions and key bindings installed.
    Type #? or press the HELP key to see keybindings"])
```

(TEDITKEY.DEINSTALL

```

[LAMBDA (ARGS |...|) (* gbn "10-Oct-85 00:04")
  (MODIFY.KEYACTIONS \ORIGKEYACTIONS)
  (SELECTQ (MACHINETYPE)
    (DANDELION (MODIFY.KEYACTIONS \DLIONKEYACTIONS))
  (PROGN NIL])
```

(\TK.ACTIONTOCHARCODE

```

[LAMBDA (FN) (* gbn "23-Feb-85 17:17")
  (* takes the name of the function and looks in TEDITKEY.KEYBINDINGS to find out which CHARCODE generates that
  behaviour)
  (PROG ((PAIR (ASSOC FN TEDITKEY.KEYBINDINGS)))
    (RETURN (if PAIR
      then [EVAL `(CHARCODE %, (CAADR PAIR]
      else NIL])
```

(\TK.BUILD.MENU

```

[LAMBDA (KEYBINDINGS) (* gbn "23-Feb-85 17:17")
  (* builds a menu to display the key bindings)
  (PROG (ITEMS)
    [for TRIPLE in TEDITKEY.KEYBINDINGS do (COND
      (TRIPLE (push ITEMS (LIST (CADR TRIPLE)
        `(QUOTE %, TRIPLE)
        "Function which is performed by the
        key(s) to the right of the mouse"))
      (push ITEMS (LIST (for DESC in (CADR TRIPLE)
        collect (CHARDESC DESC)
        NIL)))
      (T (* insert a space since NIL marks logical divisions in the list)
        (push ITEMS ' (" NIL ""))
        (push ITEMS ' (" NIL ""])
    (SETQ \TK.MENU (create MENU
      ITEMS _ (DREVERSE ITEMS)
      MENCOLUMNS _ 2
      CENTERFLG _ T
      MENUFONT _ (FONTCREATE 'HELVETICA 10))
```

(\TK.HELP

```

[LAMBDA (WHATEVER) (* gbn " 5-Nov-84 18:17")
  (* brings up a menu of the available key bindings)
```

(MENU \TK.MENU])

(\TK.SETFONTINLOOKS

[LAMBDA (TEXTSTREAM LOOKS) (* gbn "11-Oct-85 07:12")

(* * rebuilds the font field of looks according to the values in the fields)

```

(PROG (NEWFONT)
  (SETQ NEWFONT (FONTCREATE (OR (fetch CLNAME of LOOKS)
                                (FONTPROP (fetch CLFONT of LOOKS)
                                           'FAMILY))
                            (fetch CLSIZE of LOOKS)
                            (LIST (if (fetch CLBOLD of LOOKS)
                                       then 'BOLD
                                       else 'MEDIUM)
                                  (if (fetch CLITAL of LOOKS)
                                       then 'ITALIC
                                       else 'REGULAR)
                                  'REGULAR)
                            NIL NIL T))
    (if (CAR NEWFONT)
        then
          (RETURN (replace CLFONT of LOOKS with NEWFONT)) (* we got the font, so now replace it)
        else
          (TEDIT.PROMPTPRINT TEXTSTREAM (CONCAT "Font not found: "
                                                (CONCAT [L-CASE (OR (fetch CLNAME of LOOKS)
                                                                (FONTPROP (fetch CLFONT of LOOKS)
                                                                    'FAMILY)]
                                                    " "
                                                    (fetch CLSIZE of LOOKS)
                                                    (if (fetch CLBOLD of LOOKS)
                                                        then 'BOLD " bold"
                                                        else "")
                                                    (if (fetch CLITAL of LOOKS)
                                                        then " italic"
                                                        else ""))))
          (* we lost, print a msg and return NIL so that the caller knows.)
          (TEDIT.PROMPTPRINT TEXTSTREAM (CONCAT "Font not found: "
                                                (CONCAT [L-CASE (OR (fetch CLNAME of LOOKS)
                                                                (FONTPROP (fetch CLFONT of LOOKS)
                                                                    'FAMILY)]
                                                    " "
                                                    (fetch CLSIZE of LOOKS)
                                                    (if (fetch CLBOLD of LOOKS)
                                                        then 'BOLD " bold"
                                                        else "")
                                                    (if (fetch CLITAL of LOOKS)
                                                        then " italic"
                                                        else ""))))
          T)
    (RETURN NIL])

```

(WRITE.CHARDESC.AUX

[LAMBDA (TOKENS) (* gbn "10-Oct-85 00:20")

```

(COND
  ((EQ (LENGTH TOKENS)
       1)
   (CONS (CAR TOKENS)
         NIL))
  (T (SELECTQ (CAR TOKENS)
              (%# [CONS "meta " (WRITE.CHARDESC.AUX (COND
                                                       ((AND (CDR TOKENS)
                                                            (EQ (CADR TOKENS)
                                                                '%#))
                                                       (CDDR TOKENS]))
              (^ (CONS "control " (WRITE.CHARDESC.AUX (CDR TOKENS))))
              (ERROR CHARDESC " is a misunderstood character descriptor"]

```

(CHARDESC

[LAMBDA (CHARDESC) (* gbn " 7-Nov-84 14:21")

(* takes a description in the form taken as input to charcode and writes out a human readable form)

(PACK (WRITE.CHARDESC.AUX (UNPACK CHARDESC]))

(TEDITKEY.CONFIGURE

[LAMBDA NIL (* gbn " 5-Nov-84 18:58")

(PROMPTPRINT "not implemented"])

(\TK.ADDKEY

[LAMBDA (TRIPLE) (* gbn " 5-Nov-84 18:41")

]) (* dummy for now)

(\TK.CHANGEKEY

[LAMBDA (THIS) (* gbn " 5-Nov-84 18:42")

]) (* DUMMY)

(\TK.APPLYPENDING

[LAMBDA (TEXTSTREAM TEXTOBJ SEL) (* gbn "10-Dec-84 15:58")

(* * takes the entries on \TK.PENDING, reverses them and applies them as incremental changes to the selection.)

```
(PROG ((PENDING (DREVERSE \TK.PENDING))
      (LOOKS (LIST NIL)))
  (for ENTRY in PENDING do (SELECTQ ENTRY
    (BOLDON (LISTPUT LOOKS 'WEIGHT 'BOLD))
    (BOLDOFF (LISTPUT LOOKS 'WEIGHT 'MEDIUM))
    (ITALICON (LISTPUT LOOKS 'SLOPE 'ITALIC))
    (ITALICOFF (LISTPUT LOOKS 'SLOPE 'REGULAR))
    (UNDERLINEON (LISTPUT LOOKS 'UNDERLINE 'ON))
    (UNDERLINEOFF (LISTPUT LOOKS 'UNDERLINE 'OFF))
    (SUPERSCRIPT NIL) (* nothing for the moment)
    (SUBSCRIPT NIL) (* nothing for the moment)
    (LARGER NIL) (* nothing for the moment)
    (SMALLER NIL) (* nothing for the moment)
    (DEFAULTS (SETQ LOOKS (\TEDIT.UNPARSE.CHARLOOKS.LIST
                          TEDIT.DEFAULT.CHARLOOKS)))
    ((TIMESROMAN HELVETICA GACHA MODERN CLASSIC TERMINAL HIPPO MATH)
     (LISTPUT LOOKS 'FAMILY ENTRY))
    (\LISPERROR "Illegal pending operation in \TK.PENDING" ENTRY)))
  (SETQ \TK.PENDING NIL)
  (RETURN (TEDIT.LOOKS TEXTSTREAM LOOKS SEL]))
```

(\TK.NTHFONT

```
[LAMBDA (N) (* gbn "27-Jan-85 17:51")
  (* returns the name of the nth teditkey font)
  (CAR (NTH TEDITKEY.FONTN N))
)
```

:: redefinition of system junk

(DEFINEQ

(METASHIFT

```
[LAMBDA (FLG) (* gbn "6-Mar-85 15:43")
  (* Sets interpretation of TEDITKEY.METAKEY key to first arg, where T means meta-shift, NIL means original setting.
  Returns previous setting)
```

```
(PROG ((METASTATUS ' (METADOWN . METAUP))
      (OLDSETTING))
  [SETQ OLDSETTING (KEYACTION TEDITKEY.METAKEY (AND (IGREATERP FLG 0)
    (COND
      ((EQ (ARG FLG 1)
        T)
        METASTATUS)
      (T (OR (ARG FLG 1)
        (CDR (ASSOC TEDITKEY.METAKEY
          \ORIGKEYACTIONS))))))
  (RETURN (COND
    ((EQUAL OLDSETTING METASTATUS)
      T)
    (T OLDSETTING]))
)
```

:: (\TK.BOLDTOGGLE (##H ##h) 'toggle Bold caret looks') (\TK.ITALICTOGGLE (##i ##I) 'toggle Italic caret looks') (\TK.SUPERSCRIPT.CARET (##^) 'Superscript the caret looks') (\TK.SUBSCRIPT.CARET (##_) 'Subscript the caret looks')

(DEFINEQ

(TEDIT.FULL.FIND

```
[LAMBDA (TEXTSTREAM SEARCHSTRING) (* gbn "8-Mar-85 12:56")
  (PROG (TARGET (TEXTOBJ (TEXTOBJ TEXTSTREAM))
        (SEL CH W) (* Case sensitive search, with * and %# wildcards)
        [SETQ W (CAR (MKLIST (fetch \WINDOW of TEXTOBJ)
        [SETQ TARGET (OR SEARCHSTRING (TEDIT.GETINPUT TEXTOBJ "Text to find: " (WINDOWPROP W
          'TEDIT.LAST.FIND.STRING)
          (CHARCODE (EOL LF ESC)
        [COND
          (TARGET (SETQ SEL (fetch SEL of TEXTOBJ))
            (\SHOWSEL SEL NIL NIL)
            (TEDIT.PROMPTPRINT TEXTOBJ "Searching..." T)
            (SETQ CH (TEDIT.FIND TEXTOBJ (MKSTRING TARGET)
              NIL NIL T))
          (COND
            (CH (* We found the target text.)
              (TEDIT.PROMPTPRINT TEXTOBJ "Done.")
              (replace CH# of SEL with (CAR CH)) (* Set up SELECTION to be the found text)
              (replace CHLIM of SEL with (ADD1 (CADR CH)))
              (replace DCH of SEL with (ADD1 (IDIFFERENCE (CADR CH)
```

(CAR CH]

(replace POINT of SEL with 'RIGHT)
(replace CARETLOOKS of TEXTOBJ with (\TEDIT.GET.INSERT.CHARLOOKS TEXTOBJ SEL))
(TEDIT.RESET.EXTEND.PENDING.DELETE SEL)

(* And never pending a deletion.)

(\FIXSEL SEL TEXTOBJ)
(TEDIT.NORMALIZECARET TEXTOBJ)
(\SHOWSEL SEL NIL T)
(WINDOWPROP W 'TEDIT.LAST.FIND.STRING TARGET)

(* And get it into the window)

)
(T (TEDIT.PROMPTPRINT TEXTOBJ "(not found)")
(\SHOWSEL SEL NIL T)

(replace \INSERTNEXTCH of TEXTOBJ with -1])

)

(RPAQQ \TK.WHITESPACE 22)

(RPAQ TEDIT.INTERRUPTS `((%, (CHARCODE ^G)
ERROR)
(%, (CHARCODE ^C)
HELP))

(DECLARE%: EVAL@COMPILE

(RPAQQ \TK.WHITESPACE 22)

(CONSTANTS (\TK.WHITESPACE 22))
)

(RPAQ? TEDITKEY.VERBOSE T)

(RPAQ? TEDITKEY.METAKEY 'TAB)

(RPAQ? TEDITKEY.LOCKTOGGLEKEY NIL)

(RPAQ? TEDITKEY.NESTWIDTH 36)

(RPAQ? \TK.SIZEINCREMENT 2)

(RPAQ? TEDITKEY.OFFSETINCREMENT 3)

(RPAQ? TEDITKEY.FONTS '(TIMESROMAN HELVETICA GACHA MODERN CLASSIC TERMINAL SYMBOL HIPPO))

(RPAQ? TEDITKEY.FNKEYFLG T)

(DECLARE%: EVAL@COMPILE

(PUTPROPS METACODE MACRO [LAMBDA (CHARCODE)
(LOGOR CHARCODE 128)]

(PUTPROPS CONTROLCODE MACRO [LAMBDA (CHARCODE)
(LOGAND CHARCODE 31)]

(PUTPROPS LCMETACODE MACRO [LAMBDA (CHARCODE)
(LOGOR 160 CHARCODE)]

)
(RPAQ? \TK.SELKEY 'OPEN)

(RPAQ? \TK.PENDING)

(RPAQ? TEDITKEY.KEYBINDINGS

`((\TK.FONT1 (%##1)
%,
(CONCAT "change to font " (\TK.NTHFONT 1)))
(\TK.FONT2 (%##2)
%,
(CONCAT "change to font " (\TK.NTHFONT 2)))
(\TK.FONT3 (%##3)
%,
(CONCAT "change to font " (\TK.NTHFONT 3)))
(\TK.FONT4 (%##4)
%,
(CONCAT "change to font " (\TK.NTHFONT 4)))
(\TK.FONT5 (%##5)
%,
(CONCAT "change to font " (\TK.NTHFONT 5)))
(\TK.FONT6 (%##6)
%,
(CONCAT "change to font " (\TK.NTHFONT 6)))
(\TK.FONT7 (%##7)
%,
(CONCAT "change to font " (\TK.NTHFONT 7)))
(\TK.FONT8 (%##8)
%,

```

(CONCAT "change to font " (\TK.NTHFONT 8)))
NIL
(\TK.DEFAULTS.CARET (###/)
 "restore the default caret looks")
(\TK.SMALLER.CARET (###9)
 "decrease the caret font size")
(\TK.LARGER.CARET (###0)
 "increase the caret font size")
(\TK.SHOWCARETLOOKS (###=)
 "display the current caret looks")
NIL
(\TK.REDISPLAY (###R ###r)
 "Restore the display")
(\TK.HELP (###?)
 "displays the current key bindings")
NIL
(\TK.PREVCHAR (^B ^b)
 "Back one character")
(\TK.NEXTCHAR (^F ^f)
 "Forward one character")
(\TK.FORWARD.WORD (###F ###f)
 "Forward one word")
(\TK.BACK.WORD (###B ###b)
 "Back one word")
(\TK.GOTOLINEBEGIN (^A ^a)
 "go to stArt of line")
(\TK.GOTOLINEEND (^E ^e)
 "go to End of line")
(\TK.PREVLIN (^P ^p)
 "go to Previous line")
(\TK.NEXTLINE (^N ^n)
 "go to Next line")
(\TK.GOTODOCBEGIN (###<)
 "start of document")
(\TK.GOTODOCEND (###>)
 "end of document")
(\TK.SELECT.ALL (###S ###s)
 "Select whole document")
NIL
(\TK.DELLINEFORWARD (^K ^k)
 "Kill line")
(\TK.OPENLINE (^O ^o)
 "Open up blank line")
(\TK.DELCHARFORWARD (^D ^d)
 "Delete character forward")
(\TK.DEL.WORD.FORWARD (###D ###d)
 "Delete word forward")
(\TK.TRANSPOSECHARS (^T ^t)
 "Transpose characters")
NIL NIL (\TK.NEST (|##[|)
 "indents margins (nest)")
(\TK.UNNEST (|##|)
 "exdents margins (unnest)")
(\TK.CENTER.SEL (###J ###j)
 "alter Justification")
(\TK.UCASE.SEL (###U ###u)
 "Uppercasify selection")
(\TK.CAPITALISE.SEL (###C ###c)
 "Capitalize selection")
(\TK.LCASE.SEL (###L ###l)
 "Lowercasify selection")
(GET.OBJ.FROM.USER (###O ###o)
 "insert Object"))

```

```

(RPAQ? TEDITKEY.DLION.KEYACTIONS
 `((STOP (% (CHARCODE ^G)
           %,
           (CHARCODE ^C)
           NOLOCKSHIFT))
  (OPEN (% (CHARCODE 2,1)
           %,
           (CHARCODE 2,41)
           NOLOCKSHIFT))
  (FONT FONTDOWN . FONTUP)
  (KEYBOARD USERMODE1DOWN . USERMODE1UP)))

```

```

(RPAQ? COMS
 ;; (TEDITKEY.FNKEYACTIONS (BQUOTE ((CENTER (, (CHARCODE ##^B), (CHARCODE ##^C))) (BOLD (, (CHARCODE ##^D),
 ;; (CHARCODE ##^E) NOLOCKSHIFT)) (ITALICS (, (CHARCODE ##^F), (CHARCODE ##^G) NOLOCKSHIFT)) (UNDERLINE (,
 ;; (CHARCODE ##^H), (CHARCODE ##^I) NOLOCKSHIFT)) (SUPERSCRIP (, (CHARCODE ##^J), (CHARCODE ##^K)
 ;; NOLOCKSHIFT)) (SUBSCRIPT (, (CHARCODE ##^L), (CHARCODE ##^N) NOLOCKSHIFT)) (LARGER (, (CHARCODE ##^O),
 ;; (CHARCODE ##^P) NOLOCKSHIFT)) (DEFAULTS (, (CHARCODE ##^Q), (CHARCODE ##^R) NOLOCKSHIFT)) (BS (,
 ;; (CHARCODE ^H), (CHARCODE ^D) NOLOCKSHIFT))))))
 )

```

```

(RPAQ? TEDITKEY.DLION.KEYBINDINGS
 '(((\ACTION 'OPEN)
  \TK.OPENLINE)

```

```
( (\ACTION 'HELP)
  \TK.HELP)
( (\ACTION 'MARGINS)
  \TK.NEST)
( (\SHIFTACTION 'MARGINS)
  \TK.UNNEST)
( (\SHIFTACTION 'NEXT)
  GOTONEXTTTYWINDOW))
```

```
(RPAQ? TEDITKEY.FNKEYBINDINGS ' (( (\ACTION 'DEFAULTS)
  \TK.DEFAULTSSEL)
  ( (\SHIFTACTION 'DEFAULTS)
  \TK.SETDEFAULTLOOKS)))
```

```
(RPAQ? COMS ;; NOT NEEDED (TEDITKEY.DLION.KEYSYNTAX (QUOTE (( (\ACTION (QUOTE NEXT)) NEXT) (( (\ACTION (QUOTE UNDO))
  ;; UNDO) (( (\ACTION (QUOTE BS)) CHARDELETE))))
  )
```

```
(RPAQ? TEDITKEY.DORADO.KEYACTIONS
  \ ( (BS (% , (CHARCODE ^H)
    % ,
    (CHARCODE ^H)))
  (BLANK-BOTTOM (% , (CHARCODE %##^A)
    % ,
    (CHARCODE %##^A)))
  (BLANK-TOP FONTDOWN . FONTUP)
  (BLANK-MIDDLE USERMODE1DOWN . USERMODE1UP)
  (CENTER (2,101 2,141 NOLOCKSHIFT))
  (BOLD (2,102 2,142 NOLOCKSHIFT))
  (ITALICS (2,103 2,143 NOLOCKSHIFT))
  (UNDERLINE (2,106 2,146 NOLOCKSHIFT))
  (SUPERSCRIP (2,113 2,153 NOLOCKSHIFT))
  (SUBSCRIPT (2,114 2,154 NOLOCKSHIFT))
  (LARGER (2,110 2,150 NOLOCKSHIFT))
  (DEFAULTS (2,115 2,155 NOLOCKSHIFT)))
```

```
(RPAQ? TEDITKEY.DORADO.KEYSYNTAX ' (( (CHARCODE %##N)
  NEXT)
  (( (CHARCODE %##n)
  NEXT)
  (( (\ACTION 'BLANK-BOTTOM)
  UNDO)
  (( (\ACTION 'BS)
  CHARDELETE)))
```

(TEDITKEY.INSTALL)

(\TK.BUILD.MENU)

(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVERS

(ADDTOVAR NLAMA)

(ADDTOVAR NLAML)

(ADDTOVAR LAMA METASHIFT)

)

(PUTPROPS TEDITKEY COPYRIGHT ("Xerox Corporation" 1985 1986 1987))

FUNCTION INDEX

CHARDESC	17	\TK.BOLD.SEL.ON	4	\TK.GOTODOCBEGIN	11	\TK.SELECT.ALL	13
GOTONEXTTTYWINDOW	10	\TK.BOLDTOGGLE	10	\TK.GOTODOCEND	12	\TK.SETCARETFONT	9
METASHIFT	18	\TK.BUILD.MENU	16	\TK.GOTOLINEBEGIN	12	\TK.SETDEFAULTLOOKS	10
NTHCAR	3	\TK.CAPITALISE.SEL	6	\TK.GOTOLINEEND	12	\TK.SETFILEPTR.TO.CARET	14
TEDIT.FULL.FIND	18	\TK.CENTER.SEL	6	\TK.HELP	16	\TK.SETFONTINLOOKS	17
TEDITKEY.CONFIGURE	17	\TK.CENTER.SEL.REV	7	\TK.ITALIC.CARET.OFF	8	\TK.SHOWCARETLOOKS	7
TEDITKEY.DEINSTALL	16	\TK.CHANGEKEY	17	\TK.ITALIC.CARET.ON	8	\TK.SMALLER.CARET	9
TEDITKEY.INSTALL	15	\TK.DEFAULTS.CARET	9	\TK.ITALIC.SEL.OFF	4	\TK.SMALLERSEL	5
WRITE.CHARDESC.AUX	17	\TK.DEFAULTSSEL	5	\TK.ITALIC.SEL.ON	4	\TK.SUBSCRIPT.CARET	9
\ACTION	15	\TK.DEL.WORD.FORWARD	5	\TK.ITALICTOGGLE	10	\TK.SUBSCRIPTSEL	5
\CAPITALISE	6	\TK.DELCHARFORWARD	13	\TK.LARGER.CARET	9	\TK.SUPERSCRIP.T.CARET	8
\SEL.LIMIT	14	\TK.DELLINEFORWARD	13	\TK.LARGERSEL	5	\TK.SUPERSCRIP.TSEL	5
\SEL.LINEDESC	14	\TK.FIND	13	\TK.LCASE.SEL	6	\TK.TRANSPOSECHARS	14
\SHIFTACTION	15	\TK.FONT1	9	\TK.NEST	7	\TK.UCASE.SEL	6
\TEXTOBJ.WINDEX	3	\TK.FONT2	9	\TK.NEXTCHAR	12	\TK.UNDERLINE.CARET.OFF	8
\TK.ACTIONTOCHARCODE	16	\TK.FONT3	9	\TK.NEXTLINE	11	\TK.UNDERLINE.CARET.ON	8
\TK.ADDKEY	17	\TK.FONT4	9	\TK.NTHFONT	18	\TK.UNDERLINE.SEL.OFF	4
\TK.APPLYPENDING	17	\TK.FONT5	10	\TK.OPENLINE	13	\TK.UNDERLINE.SEL.ON	4
\TK.BACK.WORD	13	\TK.FONT6	10	\TK.PREVCHAR	12	\TK.UNDERLINETOGGLE	10
\TK.BOLD.CARET.OFF	8	\TK.FONT7	10	\TK.PREVLINELINE	11	\TK.UNNEST	7
\TK.BOLD.CARET.ON	8	\TK.FONT8	10	\TK.PREVSCHREEN	3		
\TK.BOLD.SEL.OFF	4	\TK.FORWARD.WORD	12	\TK.DISPLAY	13		

VARIABLE INDEX

COMS	20, 21	TEDITKEY.FNKEYBINDINGS	21	TEDITKEY.NESTWIDTH	19
TEDIT.INTERRUPTS	19	TEDITKEY.FNKEYFLG	19	TEDITKEY.OFFSETINCREMENT	19
TEDITKEY.DLION.KEYACTIONS	20	TEDITKEY.FONTS	19	TEDITKEY.VERBOSE	19
TEDITKEY.DLION.KEYBINDINGS	20	TEDITKEY.KEYBINDINGS	19	\TK.PENDING	19
TEDITKEY.DORADO.KEYACTIONS	21	TEDITKEY.LOCKTOGGLEKEY	19	\TK.SELKEY	19
TEDITKEY.DORADO.KEYSYNTAX	21	TEDITKEY.METAKEY	19	\TK.SIZEINCREMENT	19

MACRO INDEX

CONTROLCODE	19	METACODE	19	\SEL.LIMIT.FORWARD	14
LCMETACODE	19	\LINEDESC.LAST.REAL.CHAR	15	\TK.ONOROFF	15

CONSTANT INDEX

\TK.WHITESPACE	19
----------------------	----