

File created: 13-Oct-2023 11:18:04 {WMEDLEY}<lispusers>EXAMINEDEFS.;48

edit by: rmk

changes to: (FNS EXAMINEDEFS TEDITDEF)

previous date: 19-Jul-2023 13:59:26 {WMEDLEY}<lispusers>EXAMINEDEFS.;44

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

```
(RPAQQ EXAMINEDEFSCOMS ((FNS EXAMINEDEFS EXAMINEFILES TEDITDEF)
  (INITVARS (EXAMINEDEFS-PROCESS-LIST)
    (EXAMINEWITH 'COMPARETEXT))
  (FILES (SYSLOAD)
    COMPARETEXT)))
```

(DEFINEQ

(EXAMINEDEFS

```
[LAMBDA (NAME TYPE SOURCE1 SOURCE2 TITLE1 TITLE2 REGION)
; Edited 13-Oct-2023 11:11 by rmk
; Edited 18-May-2023 22:35 by rmk
; Edited 21-Apr-2023 14:42 by rmk
```

```
:: This provides for side-by-side examination of separate but presumably related expressions. The (LISTP) expressions can be provided directly as
:: the definitions SOURCE1 and SOURCE2 or, if NAME is given the copies of the definitions of NAME as TYPE on the two sources are examined.
:: If both SOURCE1 and SOURCE2 are NIL, then SOURCE1 is the existing file definitions, NIL is the existing in-memory definition
```

```
::
```

```
(if NAME
  then (CL:UNLESS [OR SOURCE1 SOURCE2 (SETQ SOURCE2 (CAR (WHEREIS NAME (OR TYPE ' (FNS FUNCTIONS))
    T]
    (ERROR (CONCAT "Can't find " NAME " definitions to examine"))))
  else (CL:UNLESS (LISTP SOURCE1)
    (ERROR SOURCE1 " cannot be examined"))
    (CL:UNLESS (LISTP SOURCE2)
    (ERROR SOURCE2 " cannot be examined"))))
```

```
:: TITLE1 and TITLE2 are optional strings that will be used to construct the titles of the SEDIT windows. We would like to know where GETDEF got
:: the definition so we can use that, but there isn't an interface that provides that information (extended WHEREIS?)
```

```
::
```

```
:: If SOURCE1 and SOURCE2 are both NIL, SOURCE1 defaults to the current (in memory) definition, SOURCE2 defaults to the definition on the
:: current file.
```

```
(LET (DEF1 DEF2)
  (if (SETQ DEF1 (LISTP SOURCE1))
    elseif TYPE
      then (NEQ (SETQ DEF1 (GETDEF NAME TYPE SOURCE1 'NOERROR))
        (FILEPKGTYPE TYPE 'NULLDEF))
    elseif (NEQ (SETQ DEF1 (GETDEF NAME (SETQ TYPE 'FNS)
      SOURCE1
      'NOERROR))
        (FILEPKGTYPE TYPE 'NULLDEF))
    elseif (NEQ (SETQ DEF1 (GETDEF NAME (SETQ TYPE 'FUNCTIONS)
      SOURCE1
      'NOERROR))
        (FILEPKGTYPE TYPE 'NULLDEF))
    else (ERROR NAME (CONCAT "not found on " SOURCE1)))
  (if (SETQ DEF2 (LISTP SOURCE2))
    elseif (NEQ (SETQ DEF2 (GETDEF NAME TYPE SOURCE2 'NOERROR))
      (FILEPKGTYPE TYPE 'NULLDEF))
    else (ERROR NAME (CONCAT "not found on " SOURCE2)))
  (CL:UNLESS TITLE1
    (SETQ TITLE1 (OR (AND (OR (LISTP SOURCE1)
      (NULL SOURCE1))
      'Current)
      (AND (MEMB (U-CASE SOURCE1)
        ' (PROP SAVED))
      'Saved)
      (FINDFILE SOURCE1)
      SOURCE1)))
  (CL:UNLESS TITLE2
    (SETQ TITLE2 (OR (AND (OR (LISTP SOURCE2)
      (NULL SOURCE2))
      'Current)
      (AND (MEMB (U-CASE SOURCE2)
        ' (PROP SAVED))
      'Saved)
      (FINDFILE SOURCE2)
      SOURCE2)))
  (SELECTQ (EDITMODE)
    (SEDT:SEDT
```

```
:: A kludge to eliminate dangling SEDIT processes from previous examinations
[SETQ EXAMINEDEFS-PROCESS-LIST (FOR PAIR IN EXAMINEDEFS-PROCESS-LIST
  COLLECT (IF (OPENWP (CAR PAIR))
```

```

                                THEN PAIR
                                ELSE (DEL.PROCESS (CDR PAIR))
                                      (GO $$ITERATE]

;; Set it up for new side-by-side regions that are forgotten when the window is closed. Their shape is usually not that
;; useful for regular edits.
;; Crude suggestions for height, width, position. Suggest shorter window for smaller structures
(SELECTQ EXAMINewith
  (SEdit (SETQ DEF1 (COPY DEF1)) ; Copy to simulate read-only
    (SETQ DEF2 (COPY DEF2))
    (CL:UNLESS (REGIONP REGION)
      (SETQ REGION (GETREGION)))
    [LET (R1 R2 HALFWIDTH W1 W2)
      (SETQ HALFWIDTH (IQUOTIENT (FETCH (REGION WIDTH) OF REGION)
        2))
      (SETQ R1 (CREATE REGION USING REGION WIDTH _ HALFWIDTH))
      (SETQ R2 (CREATE REGION USING REGION LEFT
        (IPLUS (FETCH (REGION LEFT)
          OF REGION)
          HALFWIDTH)
        WIDTH _ HALFWIDTH))

      [SETQ W1
        (SEdit:GET-WINDOW (SEdit:SEdit DEF1
          `(:NAME ,(CONCAT NAME " from " TITLE1)
            :REGION
            , (CREATE REGION
              USING REGION WIDTH _ HALFWIDTH)
              R1 :DONT-KEEP-WINDOW-REGION T]
        [SETQ W2 (SEdit:GET-WINDOW (SEdit:SEdit
          DEF2
          `(:NAME ,(CONCAT NAME " from " TITLE2)
            :REGION
            , R2 :DONT-KEEP-WINDOW-REGION T]

        (ATTACHWINDOW W2 W1 'RIGHT 'JUSTIFY)
        (MODERNWINDOW W2)

        ;; So we can kill the processes on the next call, if they still exist after the windows are closed.
        (PUSH EXAMINEDEFS-PROCESS-LIST (CONS W1 (WINDOWPROP W1 'PROCESS))
          (CONS W2 (WINDOWPROP W2 'PROCESS)))
        (COMPARETEXT [LET (COMPARETEXT.ALLCHUNKS CTWINDOW
          (KEY (LIST NAME TYPE SOURCE1 SOURCE2 TITLE1 TITLE2))
          (TEXTWIDTH 700)
          (TEXTHEIGHT 600))
          (DECLARE (SPECVARS COMPARETEXT.ALLCHUNKS))
          ; Reuse an existing CT graph window for this DEF
          (OR [FIND W IN (OPENWINDOWS)
            SUCHTHAT (EQUAL KEY (WINDOWPROP W 'EXAMINEDEFS)
              (PROG1 (SETQ CTWINDOW
                (COMPARETEXT (TEDITDEF NAME DEF1 TYPE NIL TEXTWIDTH
                  )
                  (TEDITDEF NAME DEF2 TYPE NIL TEXTWIDTH)
                  'LINE
                  (OR REGION (GETPOSITION))
                  (LIST TITLE1 TITLE2)
                  (CONCAT "Compare sources of " NAME " as "
                    TYPE)
                  TEXTWIDTH TEXTHEIGHT))
                (WINDOWPROP CTWINDOW 'EXAMINEDEFS KEY))])
              (SHOULDNT)))
          (PROGN (EDITE DEF1)
            (EDITE DEF2])

```

(EXAMINEFILES

```

[LAMBDA (FILE1 FILE2 TITLE1 TITLE2 REGION)

```

```

; Edited 19-Jul-2023 13:48 by rmk
; Edited 1-Feb-2022 23:15 by rmk
; Edited 25-Jan-2022 10:08 by rmk
; Edited 2-Jan-2022 23:15 by rmk
; Edited 30-Dec-2021 21:49 by rmk

```

```

;; We get a region, then split it in half.

```

```

(CL:UNLESS REGION
  (SETQ REGION (GETREGION)))
(LIST (AND (INFILEP FILE1)
  (TEDIT-SEE FILE1 (RELCREATEREGION `(:,REGION 0.5 -1)
    REGION
    'RIGHT
    'TOP
    `(:,REGION 0.5)
    (FETCH (REGION TOP) OF REGION))
    NIL TITLE1))
  (AND (INFILEP FILE2)
    (TEDIT-SEE FILE2 (RELCREATEREGION `(:,REGION 0.5 1)
      REGION
      'LEFT
      'TOP
      `(:,REGION 0.5)

```

(FETCH (REGION TOP) OF REGION))

NIL TITLE2])

(TEDITDEF

[LAMBDA (NAME DEF TYPE READERENVIRONMENT WIDTH)

; Edited 13-Oct-2023 00:23 by rmk

; Edited 23-Jun-2022 17:27 by rmk

; Edited 28-Jan-2022 23:36 by rmk

; Edited 12-Jan-2022 17:27 by rmk

(LET ((TSTREAM (OPENTEXTSTREAM)))

(DSPFONT DEFAULTFONT TSTREAM)

(CL:WHEN WIDTH

(LINELENGTH (IQUOTIENT WIDTH (CHARWIDTH (CHARCODE SPACE)
TSTREAM)))

TSTREAM))

(SELECTQ (CAR DEF)

([LAMBDA NLAMBDA OPENNLAMBDA]

(PRINTOUT TSTREAM .FONT BOLDFONT .P2 NAME T .FONT DEFAULTFONT 2)

(PRINTDEF DEF 2 T NIL NIL TSTREAM))

(DEFINEQ (SETQ DEF (CADR DEF))

(PRINTOUT TSTREAM .FONT BOLDFONT .P2 NAME T .FONT DEFAULTFONT 2)

(PRINTDEF (CADR DEF)

2 T NIL NIL TSTREAM))

((DEFMACRO DEFUN DEFMACRO CL:DEFUN)

; Has args after name

(PRINTOUT TSTREAM "(" .P2 (CAR DEF)

" " .FONT BOLDFONT .P2 (CADR DEF)

.FONT DEFAULTFONT " " .P2 (CADDR DEF)

T)

(PRINTDEF (CDDDR DEF)

3 T T NIL TSTREAM)

(PRIN3 ")" TSTREAM))

(IF (EQ NAME (CADR DEF))

THEN ;; Like RPAQQ, bold the name

[PRINTOUT TSTREAM "(" .P2 (CAR DEF)

" " .FONT BOLDFONT .P2 (CADR DEF)

.FONT DEFAULTFONT T .TAB (IPLUS 2 (NCHARS (CAR DEF]

(PRINTDEF (CDDR DEF)

(IPLUS 2 (NCHARS (CAR DEF)))

T T NIL TSTREAM)

(PRIN3 ")" TSTREAM)

ELSE (PRINTDEF DEF 3 NIL NIL NIL TSTREAM))

TSTREAM])

)

(RPAQ? **EXAMINEDEFS-PROCESS-LIST**)(RPAQ? **EXAMINewith** 'COMPARETEXT)(FILESLOAD (SYSLOAD)
COMPARETEXT)

FUNCTION INDEX

EXAMINEDEFS1 EXAMINEFILES2 TEDITDEF3

VARIABLE INDEX

EXAMINEDEFS-PROCESS-LIST3 EXAMINEWITH3
