

File created: 22-Jul-88 15:32:46 {MCS:MCS:STANFORD}<LANE>SKETCHTALK.;26

previous date: 13-Jun-88 16:34:08 {MCS:MCS:STANFORD}<LANE>SKETCHTALK.;25

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::  
:: Copyright (c) 1987, 1988 by Stanford University. All rights reserved.

(RPAQQ **SKETCHTALKCOMS**

```
(( * TALK Sketch Service)
 (LOCALVARS . T)
 (FNS TALK.SKETCH.DISPLAY TALK.SKETCH.LISTEN)
 (FNS TALK.SKETCH.FIND.ELEMENT TALK.SKETCH.FIND.SYMBOLS)
 (* Sketch Viewer Control Properties)
 (FNS TALK.SKETCH.WHENADDED TALK.SKETCH.WHENCHANGED TALK.SKETCH.WHENDELETED TALK.SKETCH.WHENMOVED TALK.SKETCH.PREMOVED)
 (FNS TALK.SKETCH.WHENGROUPED TALK.SKETCH.WHENUNGROUPED)
 (VARS TALK.TO.SKETCH.PROPS TALK.SKETCH.REDISPLAY.PROPS)
 (* TALK Sketch Actions)
 (FNS TALK.SKETCH.ADD.ELEMENT TALK.SKETCH.CHANGE.ELEMENT TALK.SKETCH.DELETE.ELEMENTS TALK.SKETCH.MOVE.ELEMENTS TALK.SKETCH.POSITION.ELEMENTS)
 (VARS TALK.SKETCH.ACTIONS)
 (* TALK Sketch Data)
 (VARS TALK.SKETCH.DELETE.ITEMS)
 (INITVARS TALK.SKETCH.TRACK)
 (GLOBALVARS TALK.SKETCH.ACTIONS TALK.TO.SKETCH.PROPS TALK.SKETCH.REDISPLAY.PROPS TALK.SKETCH.DELETE.ITEMS TALK.SKETCH.TRACK)
 (* etc)
 (FILES TALK.SKETCH)
 (APPENDVARS (GAP.SERVICETYPES (7 Sketch TALK.NS.SERVER))
 (TALK.SERVICETYPES (Sketch TALK.SKETCH.DISPLAY TALK.SKETCH.LISTEN)))
 (* Sketch Bug Fixes)
 (FNS TALK.SKETCH.NOP)
 (P (CHANGENAME '\SK.PUT.FONT 'SK.INSURE.TEXT 'TALK.SKETCH.NOP))
 (ADVISE BITMAPELT.CHANGEFN SK.IMAGEOBJ.CHANGEFN (TEXTUREP :IN SKFILLINGP))))
```

(\* TALK Sketch Service)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(LOCALVARS . T)
)

(DEFINEQ

(**TALK.SKETCH.DISPLAY**

```
[LAMBDA (MAINWINDOW WINDOW INPUTSTREAM OUTPUTSTREAM PROTOCOL USER)
; Edited 9-Jun-88 16:36 by cdl
(LET (MENUWINDOW)
 (SKETCH NIL MAINWINDOW)
 (SKETCH NIL WINDOW)
 (DETACHWINDOW (SETQ MENUWINDOW (WINDOWPROP WINDOW 'SKETCHFIXEDMENU NIL)))
 (CLOSEW MENUWINDOW)
 (DETACHWINDOW (SETQ MENUWINDOW (WINDOWPROP MAINWINDOW 'SKETCHFIXEDMENU NIL)))
 (CLOSEW MENUWINDOW)
 (WINDOWPROP MAINWINDOW 'SKETCHFIXEDMENU (ATTACHMENU (LET ((ITEMS (SKETCH.COMMANDMENU.ITEMS NIL T)))
 (for key in TALK.SKETCH.DELETE.ITEMS
 do (setq ITEMS (DREMOVE (SASSOC KEY ITEMS)
 ITEMS)))
 (SKETCH.COMMANDMENU ITEMS))
 MAINWINDOW
 'RIGHT
 'TOP))
 (WINDOWPROP MAINWINDOW 'SKETCHPOPUPMENU NIL)
 (WINDOWPROP WINDOW 'SKETCHPOPUPMENU NIL)
 (for pair on TALK.TO.SKETCH.PROPS do (PUTSKETCHPROP MAINWINDOW (CAR PAIR)
 (CADR PAIR)))
 (PUTSKETCHPROP MAINWINDOW 'TALK OUTPUTSTREAM) (* Still need to combine the two prompt windows into one)
 (WINDOWPROP MAINWINDOW 'SCROLLFN NIL)
 (WINDOWPROP WINDOW 'SCROLLFN NIL)
 (PUTWINDOWPROP MAINWINDOW 'DONTQUERYCHANGES T)
 (PUTWINDOWPROP WINDOW 'DONTQUERYCHANGES T)
 (RPLACA (CDAR (INSURE.SKETCH MAINWINDOW))
 (CONCAT "Talk with " USER))
 (WINDOWDELPROP WINDOW 'SHRINKFN 'SK.SHRINK.ICONCREATE)
 (WINDOWDELPROP WINDOW 'SHRINKFN 'SK.RETURN.TTY)
 (with REGION (DSPCLIPPINGREGION NIL MAINWINDOW)
 (SKED.SET.SELECTION (CREATEPOSITION (QUOTIENT WIDTH 2)
 (QUOTIENT HEIGHT 2))
 MAINWINDOW))
```

(TTY.PROCESS (WINDOWPROP MAINWINDOW 'PROCESS])

(TALK.SKETCH.LISTEN

```

[LAMBDA (MAINWINDOW WINDOW INPUTSTREAM OUTPUTSTREAM PROTOCOLTYPE)
; Edited 7-Jun-88 08:46 by cdl
(* DECLARATIONS%: (RECORD EXPR
(KEY . ARGUMENTS)) (RECORD OPERATION
(KEY FUNCTION)))
(PROG [OPERATION (EVENTFN (with TALK.PROTOCOLTYPE PROTOCOLTYPE TALK.EVENTFN))
(SKETCH (INSURE.SKETCH (MAINWINDOW WINDOW)
(DECLARE (GLOBALVARS TALK.CLOSED.STRING)
(while (OPENWP WINDOW) do (APPLY* EVENTFN INPUTSTREAM OUTPUTSTREAM)
(if (NOT (AND (OPENP INPUTSTREAM)
(OPENP OUTPUTSTREAM)))
then (RETURN)
[SELCHARQ (PEEKCCODE INPUTSTREAM)
(^G (TALK.RINGBELLS WINDOW))
(with EXPR (HREAD INPUTSTREAM)
(if (SETQ OPERATION (ASSOC KEY TALK.SKETCH.ACTIONS))
then (with OPERATION OPERATION (APPLY FUNCTION (CONS WINDOW
ARGUMENTS))
)
else (PRINTOUT (GETPROMPTWINDOW MAINWINDOW)
"Unknown Sketch Talk operation:" %, KEY]
(BIN INPUTSTREAM))
(RPLACA (CDAR SKETCH)
(CONCAT (CADAR SKETCH)
TALK.CLOSED.STRING))
(PUTSKETCHPROP MAINWINDOW 'TALK NIL])
)
)
(DEFINEQ
(TALK.SKETCH.FIND.ELEMENT
[LAMBDA (SKETCH SYMBOLS) ; Edited 18-Jun-87 09:21 by cdl
(DECLARE (SPECVARS SYMBOLS))
(SKETCH.LIST.OF.ELEMENTS SKETCH (FUNCTION (LAMBDA (ELEMENT)
(EQMEMB (GETSKETCHELEMENTPROP ELEMENT 'TALK)
SYMBOLS]))
(TALK.SKETCH.FIND.SYMBOLS
[LAMBDA (SKETCH ELEMENTS) ; Edited 18-Jun-87 11:11 by cdl
(for ELEMENT in ELEMENTS collect (GETSKETCHELEMENTPROP ELEMENT 'TALK])
)
)
(** Sketch Viewer Control Properties)
(DEFINEQ
(TALK.SKETCH.WHENADDED FN
[LAMBDA (VIEWER ELEMENT) ; Edited 23-Jun-87 07:48 by cdl
(LET [(STREAM (GETSKETCHPROP VIEWER 'TALK)
(if (AND STREAM (OPENP STREAM))
then (PROG [(SCRATCHSTREAM (OPENSTREAM '{NODIRCORE} 'BOTH)
(PUTSKETCHELEMENTPROP ELEMENT 'TALK (GENSYM 'TALK))
(HPRINT `(ADD ,ELEMENT)
SCRATCHSTREAM)
(SETFILEPTR SCRATCHSTREAM 0)
(COPYBYTES SCRATCHSTREAM STREAM)
(FORCEOUTPUT STREAM)
(CLOSEF? SCRATCHSTREAM)])
(TALK.SKETCH.WHENCHANGED FN
[LAMBDA (VIEWER ELEMENT PROPERTY NEWVALUE OLDVALUE) ; Edited 10-Jun-88 09:17 by cdl
(PROG [(STREAM (GETSKETCHPROP VIEWER 'TALK)
(if (AND STREAM (OPENP STREAM))
then (SELECTQ PROPERTY
(HASBOX (TALK.SKETCH.WHENDELETEDFN VIEWER (LIST OLDVALUE))
(TALK.SKETCH.WHENADDED FN VIEWER NEWVALUE)
(RETURN))
(DATA (SELECTQ NEWVALUE
(NIL CHANGED)
(SETQ NEWVALUE OLDVALUE))
NIL))
NIL)
(LET [(SCRATCHSTREAM (OPENSTREAM '{NODIRCORE} 'BOTH)
(HPRINT `(CHANGE ,(TALK.SKETCH.FIND.SYMBOLS VIEWER (CONS ELEMENT))
,PROPERTY
,NEWVALUE)
SCRATCHSTREAM)
(SETFILEPTR SCRATCHSTREAM 0)

```

(COPYBYTES SCRATCHSTREAM STREAM)
(FORCEOUTPUT STREAM)
(CLOSEP? SCRATCHSTREAM))

(TALK.SKETCH.WHENDELETEDFN

; Edited 23-Jun-87 07:48 by cdl

[LAMBDA (VIEWER ELEMENTS)
(PROG [(STREAM (GETSKETCHPROP VIEWER 'TALK)
(if (AND STREAM (OPENP STREAM))
then (HPRINT `(DELETE , (TALK.SKETCH.FIND.SYMBOLS VIEWER ELEMENTS))
STREAM)
(FORCEOUTPUT STREAM))

(TALK.SKETCH.WHENMOVEDFN

; Edited 23-Jun-87 10:14 by cdl

[LAMBDA (VIEWER ELEMENTS DELTA)
(PROG [(STREAM (GETSKETCHPROP VIEWER 'TALK)
(if (AND STREAM (OPENP STREAM))
then [SETQ ELEMENTS
(if (EQ (CAR ELEMENTS)
T)
then [if (NULL TALK.SKETCH.TRACK)
then (if (LISTP (CAADR ELEMENTS))
then [for ELEMENT in (CDR ELEMENTS)
collect (CONS T (TALK.SKETCH.FIND.SYMBOLS VIEWER (LIST ELEMENT])
(\* Fix for Sketch UNDO/MOVE bug)
else
(with POSITION DELTA (SETQ XCOORD (MINUS XCOORD))
(SETQ YCOORD (MINUS YCOORD)))
(LIST (CONS T (TALK.SKETCH.FIND.SYMBOLS VIEWER (CONS (CDR ELEMENTS])
elseif (in (CAR ELEMENTS) always NUMBERP)
then [LIST (CONS (CAR ELEMENTS)
(TALK.SKETCH.FIND.SYMBOLS VIEWER (LIST (CDR ELEMENTS])
else (for ELEMENT in ELEMENTS when (OR (NEQ (CAR ELEMENT)
T)
(NOT TALK.SKETCH.TRACK))
collect (CONS (CAR ELEMENT)
(TALK.SKETCH.FIND.SYMBOLS VIEWER (CONS (CDR ELEMENT])
(HPRINT `(MOVE ,ELEMENTS ,DELTA)
STREAM)
(FORCEOUTPUT STREAM))

(TALK.SKETCH.PREMOVEDFN

; Edited 23-Jun-87 07:53 by cdl

[LAMBDA (VIEWER ELEMENTS ALIGNHOW)
(LET [(STREAM (GETSKETCHPROP VIEWER 'TALK)
(if (AND TALK.SKETCH.TRACK (NULL ALIGNHOW)
(EQ (CAR ELEMENTS)
T)
STREAM
(OPENP STREAM))
then (LET [(SYMBOLS (TALK.SKETCH.FIND.SYMBOLS VIEWER (CDR ELEMENTS])
(SKETCH.TRACK.ELEMENTS (CDR ELEMENTS)
VIEWER
[FUNCTION (LAMBDA (POSITION VIEWER STREAM)
(HPRINT `(POSITION ,SYMBOLS ,POSITION)
STREAM)
(FORCEOUTPUT STREAM)
NIL NIL STREAM])

(DEFINEQ

(TALK.SKETCH.WHENGROUPEDFN

; Edited 18-Jun-87 11:02 by cdl

[LAMBDA (VIEWER ELEMENTS)
'DON'T])

(TALK.SKETCH.WHENUNGROUPEDFN

; Edited 18-Jun-87 11:02 by cdl

[LAMBDA (VIEWER ELEMENTS)
'DON'T])

(RPAQQ TALK.TO.SKETCH.PROPS (WHENADDED FN TALK.SKETCH.WHENADDED FN WHENDELETED FN TALK.SKETCH.WHENDELETED FN
WHENMOVED FN TALK.SKETCH.WHENMOVED FN WHENCHANGED FN TALK.SKETCH.WHENCHANGED FN
WHENGROUPEDFN TALK.SKETCH.WHENGROUPEDFN WHENUNGROUPEDFN
TALK.SKETCH.WHENUNGROUPEDFN PREMOVED FN TALK.SKETCH.PREMOVED FN))

(RPAQQ TALK.SKETCH.REDISPLAY.PROPS ((TEXT FONT)
(TEXTBOX FONT BRUSH)
(CLOSEDWIRE DASHING)))

(\* \* TALK Sketch Actions)

(DEFINEQ

**(TALK.SKETCH.ADD.ELEMENT**

```
[LAMBDA (SKETCH ELEMENT)
  (SKETCH.ADD.ELEMENT ELEMENT SKETCH)]
```

; Edited 21-Jun-87 11:24 by cdl

**(TALK.SKETCH.CHANGE.ELEMENT**

```
[LAMBDA (SKETCH ELEMENT PROPERTY VALUE)
```

; Edited 10-Jun-88 09:35 by cdl  
(\* DECLARATIONS%: (RECORD ENTRY  
(TYPE . PROPERTIES)))

```
(bind ENTRY for ELEMENT in (TALK.SKETCH.FIND.ELEMENT SKETCH ELEMENT)
  do (PUTSKETCHELEMENTPROP ELEMENT PROPERTY VALUE SKETCH)
    (if (SETQ ENTRY (ASSOC (SKETCH.ELEMENT.TYPE ELEMENT)
      TALK.SKETCH.REDISPLAY.PROPS)
      then (with ENTRY ENTRY (if (OR (NULL PROPERTIES)
        (MEMB PROPERTY PROPERTIES))
          then (REDISPLAYW SKETCH]))
```

**(TALK.SKETCH.DELETE.ELEMENTS**

```
[LAMBDA (SKETCH ELEMENTS)
```

; Edited 18-Jun-87 09:47 by cdl

```
(for ELEMENT inside (TALK.SKETCH.FIND.ELEMENT SKETCH ELEMENTS) do (SKETCH.DELETE.ELEMENT ELEMENT SKETCH])
```

**(TALK.SKETCH.MOVE.ELEMENTS**

```
[LAMBDA (SKETCH ELEMENTS DELTA)
```

; Edited 18-Jun-87 17:48 by cdl

```
(for PAIR in ELEMENTS do (SELECTQ (CAR PAIR)
  (T (SKETCH.MOVE.ELEMENTS (TALK.SKETCH.FIND.ELEMENT SKETCH (CDR PAIR))
    DELTA SKETCH))
  (bind POSITIONS POSITION CONTROLPT [ELEMENT _ (CAR (TALK.SKETCH.FIND.ELEMENT
    SKETCH (CDR PAIR))
  for NUMBER in (CAR PAIR)
  do (SELECTQ NUMBER
    ((1 2 3)
     (SETQ CONTROLPT (SELECTQ NUMBER
      (1 '1STCONTROLPT)
      (2 '2NDCONTROLPT)
      (3 '3RDCONTROLPT)
      (SHOULDNT)))
     (with POSITION (SETQ POSITION (COPY (GETSKETCHELEMENTPROP ELEMENT
      CONTROLPT)))
       (add XCOORD (fetch (POSITION XCOORD) of DELTA))
       (add YCOORD (fetch (POSITION YCOORD) of DELTA)))
     (PUTSKETCHELEMENTPROP ELEMENT CONTROLPT POSITION SKETCH))
  (if [SETQ POSITIONS (COPY (GETSKETCHELEMENTPROP ELEMENT 'DATA)
    then (with POSITION (CAR (NTH POSITIONS NUMBER))
      (add XCOORD (fetch (POSITION XCOORD) of DELTA))
      (add YCOORD (fetch (POSITION YCOORD) of DELTA)))
    (PUTSKETCHELEMENTPROP ELEMENT 'DATA POSITIONS SKETCH]))
```

**(TALK.SKETCH.POSITION.ELEMENTS**

```
[LAMBDA (SKETCH SYMBOLS POSITION)
```

; Edited 19-Jun-87 09:17 by cdl

```
(LET ((ELEMENTS (TALK.SKETCH.FIND.ELEMENT SKETCH SYMBOLS)))
  (SKETCH.MOVE.ELEMENTS ELEMENTS (with POSITION (GETSKETCHELEMENTPROP (CAR ELEMENTS)
    '1STCONTROLPT)
  (create POSITION
    XCOORD _ (DIFFERENCE (fetch (POSITION XCOORD) of POSITION)
      XCOORD)
    YCOORD _ (DIFFERENCE (fetch (POSITION YCOORD) of POSITION)
      YCOORD)))
  SKETCH])
```

**(RPAQQ TALK.SKETCH.ACTIONS**

```
((ADD TALK.SKETCH.ADD.ELEMENT)
 (DELETE TALK.SKETCH.DELETE.ELEMENTS)
 (MOVE TALK.SKETCH.MOVE.ELEMENTS)
 (CHANGE TALK.SKETCH.CHANGE.ELEMENT)
 (POSITION TALK.SKETCH.POSITION.ELEMENTS)))
```

(\* TALK Sketch Data)

**(RPAQQ TALK.SKETCH.DELETE.ITEMS** (Group UnGroup Put "Move view"))

**(RPAQ? TALK.SKETCH.TRACK** NIL)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS TALK.SKETCH.ACTIONS TALK.TO.SKETCH.PROPS TALK.SKETCH.REDISPLAY.PROPS TALK.SKETCH.DELETE.ITEMS  
TALK.SKETCH.TRACK)

(\* etc)

(FILESLOAD TALK SKETCH)

(APPENDTOVAR GAP.SERVICETYPES (7 Sketch TALK.NS.SERVER))

(APPENDTOVAR TALK.SERVICETYPES (Sketch TALK.SKETCH.DISPLAY TALK.SKETCH.LISTEN))

(\* \* Sketch Bug Fixes)

(DEFINEQ

**(TALK.SKETCH.NOP**

[LAMBDA (X)  
X])

; Edited 19-Jun-87 07:50 by cdl

)

(CHANGENAME '\SK.PUT.FONT 'SK.INSURE.TEXT 'TALK.SKETCH.NOP)

[XCL:REINSTALL-ADVICE 'BITMAPELT.CHANGEFN :AFTER '(:LAST (RPLACA (CDDAR (CADAR !VALUE))  
(CADDAR (CAAR !VALUE]

[XCL:REINSTALL-ADVICE 'SK.IMAGEOBJ.CHANGEFN :AFTER '(:LAST (RPLACA (CDDAR (CADAR !VALUE))  
(CADDAR (CAAR !VALUE]

[XCL:REINSTALL-ADVICE '(TEXTUREP :IN SKFILLINGP)  
:BEFORE  
'(:LAST (IF (NULL OBJECT)  
**THEN** (RETURN T]

(READWISE BITMAPELT.CHANGEFN SK.IMAGEOBJ.CHANGEFN (TEXTUREP :IN SKFILLINGP))

(PUTPROPS **SKETCHTALK COPYRIGHT** ("Stanford University" 1987 1988))

---

**FUNCTION INDEX**

TALK.SKETCH.ADD.ELEMENT . . . . .	4	TALK.SKETCH.LISTEN . . . . .	2	TALK.SKETCH.WHENCHANGEDFN . . . . .	2
TALK.SKETCH.CHANGE.ELEMENT . . . . .	4	TALK.SKETCH.MOVE.ELEMENTS . . . . .	4	TALK.SKETCH.WHENDELETEDFN . . . . .	3
TALK.SKETCH.DELETE.ELEMENTS . . . . .	4	TALK.SKETCH.NOP . . . . .	5	TALK.SKETCH.WHENGROUPEDFN . . . . .	3
TALK.SKETCH.DISPLAY . . . . .	1	TALK.SKETCH.POSITION.ELEMENTS . . . . .	4	TALK.SKETCH.WHENMOVEDFN . . . . .	3
TALK.SKETCH.FIND.ELEMENT . . . . .	2	TALK.SKETCH.PREMOVEDFN . . . . .	3	TALK.SKETCH.WHENUNGROUPEDFN . . . . .	3
TALK.SKETCH.FIND.SYMBOLS . . . . .	2	TALK.SKETCH.WHENADDEDFN . . . . .	2		

---

**VARIABLE INDEX**

TALK.SKETCH.ACTIONS . . . . .	4	TALK.SKETCH.REDISPLAY.PROPS . . . . .	3	TALK.TO.SKETCH.PROPS . . . . .	3
TALK.SKETCH.DELETE.ITEMS . . . . .	4	TALK.SKETCH.TRACK . . . . .	4		

---

**ADVICE INDEX**

BITMAPELT.CHANGEFN . . . . .	5	SK.IMAGEOBJ.CHANGEFN . . . . .	5	(TEXTUREP :IN SKFILLINGP) . . . . .	5
------------------------------	---	--------------------------------	---	-------------------------------------	---

---