File created: 24-Aug-88 08:59:17 **{XDE:MCS:STANFORD}<LISPUSERS>MEDLEY>GRAPHCALLS.;1**

changes to: (FNS GRAPHCALLS.BREAKIN GRAPHCALLS.WHEREIS)

previous date: 21-Jul-88 08:44:32 {MCS:MCS:STANFORD}<LANE>GRAPHCALLS.;36

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

```
(RPAQQ GRAPHCALLSCOMS
       [(* * GRAPHCALLS Dynamic Function Graphing)
        (LOCALVARS . T)
        (FNS GRAPHCALLS)
        (FNS GRAPHCALLS1 GRAPHCALLS.SEARCH GRAPHCALLS.ADVISE GRAPHCALLS.ADVISE1 GRAPHCALLS.BREAKIN
             GRAPHCALLS.LEFT GRAPHCALLS.MIDDLE GRAPHCALLS.COLLECT GRAPHCALLS.INSPECT.FRAME GRAPHCALLS.INSPECT
             GRAPHCALLS.INVERT.NODE GRAPHCALLS.FETCH GRAPHCALLS.STORE GRAPHCALLS.PRINT GRAPHCALLS.CLOSE
             GRAPHCALLS.GRAPH.CLOSEFN NO\ GRAPHCALLS.INSPECTCODE GRAPHCALLS.WHEREIS GRAPHCALLS.ARGLIST)
        [INITVARS GRAPHCALLS.INSPECTCODE.WINDOW
               (GRAPHCALLS.DEFAULT.OPTIONS '(:DELAY 500 :SUBFNDEFFLG T :SEARCHFN GRAPHCALLS.SEARCH :DEPTH 2
                                                   :LEFTBUTTONFN GRAPHCALLS.LEFT :MIDDLEBUTTONFN
                                                   GRAPHCALLS.MIDDLE :INSPECTWIDTH 250 :INSPECTCODEWIDTH 400
                                                   :FONT (GACHA 8)
                                                   :FORMAT
                                                   (HORIZONTAL COMPACT REVERSE/DAUGHTERS]
        (PROP ARGNAMES GRAPHCALLS)
        (PROP MENU * GRAPHCALLS.MENUS)
        (INITVARS * GRAPHCALLS.MENUS)
        (GLOBALVARS * GRAPHCALLS.MENUS)
        (GLOBALVARS GRAPHCALLS.DEFAULT.OPTIONS GRAPHCALLS.INSPECTCODE.WINDOW)
        (RECORDS GRAPHCALLS.RECORD GRAPHCALLS.OPTIONS)
        (GLOBALVARS DEFAULT.GRAPH.NODEBORDER DEFAULT.GRAPH.NODEFONT DEFAULT.GRAPH.NODELABELSHADE)
        (BLOCKS (GRAPHCALLS (SPECVARS GRAPHCALLS.SEEN)
                       (ENTRIES GRAPHCALLS)
                       GRAPHCALLS GRAPHCALLS1 GRAPHCALLS.ADVISE GRAPHCALLS.ADVISE1)
               (GRAPHCALLS.INSPECT.FRAME (SPECVARS GRAPHCALLS.COLLECTED)
                      (ENTRIES GRAPHCALLS.INSPECT.FRAME)
                      GRAPHCALLS.INSPECT GRAPHCALLS.INSPECT.FRAME GRAPHCALLS.COLLECT))
        (FILES (SYSLOAD FROM LISPUSERS)
               MSANALYZE GRAPHER)
        [P (for MENU in GRAPHCALLS.MENUS do (SET MENU (EVAL (GETPROP MENU 'MENU]
        (* * GRAPHCALLS Command Window)
        (FNS GRAPHCALLSW)
        (FNS GRAPHCALLSW.CLEAR GRAPHCALLSW.DOIT GRAPHCALLSW.FILTER GRAPHCALLSW.INCLUDE GRAPHCALLSW.PRINTFN)
        (INITVARS GRAPHCALLSW.WINDOW (GRAPHCALLSW.SCRATCHMENU (create MENU)))
        (VARS GRAPHCALLSW.MENUS)
        (PROP MENU * (PROGN GRAPHCALLSW.MENUS))
        (GLOBALVARS GRAPHCALLSW.WINDOW GRAPHCALLSW.SCRATCHMENU GRAPHCALLSW.MENUS)
        (DECLARE%: DONTCOPY (RECORDS GRAPHCALLSW.MENUS.RECORD))
        (ADDVARS (UNSAFE.TO.MODIFY.FNS ERROR ERRORX RAID RECLAIM \ALLOCBLOCK \MOVEBYTES \MP.ERROR \STOP.DRIBBLE?
                        ))
        (ALISTS (BackgroundMenuCommands GraphCalls))
        (VARS (BackgroundMenu))
        (* * Multiple Selection Menus)
        (FNS MMENU MMENU.SELECTEDFN MMENU.MARKITEM MMENU.BOLDITEM)
        (DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS (ADDVARS (NLAMA)
                                                                             (NLAML)
                                                                             (LAMA GRAPHCALLS])


          (* * GRAPHCALLS Dynamic Function Graphing)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(LOCALVARS . T)
)

(DEFINEQ

(GRAPHCALLS
  [CL:LAMBDA (FN &REST OPTIONS &KEY DEPTH &ALLOW-OTHER-KEYS)      ; Edited 21-Jul-88 08:39 by cdl
         (PROG (GRAPHNODES GRAPH GRAPHCALLS.SEEN)
               (DECLARE (SPECVARS GRAPHCALLS.SEEN))
               (if OPTIONS
                   then (for OPTION on GRAPHCALLS.DEFAULT.OPTIONS by (CDDR OPTION)
                          unless (LISTGET OPTIONS (CAR OPTION)) do (LISTPUT OPTIONS (CAR OPTION)
                                                                           (CADR OPTION)))
                 else (SETQ OPTIONS (COPY GRAPHCALLS.DEFAULT.OPTIONS)))
               (with GRAPHCALLS.OPTIONS OPTIONS (if (NULL (APPLY* :SEARCHFN FN))
                                                   then (RETURN))
                    (if :STREAM
                          then (SETQ :FONT (DSPFONT NIL :STREAM)))
                      (SETQ GRAPHNODES (GRAPHCALLS1 FN :DEPTH (FONTCOPY :FONT 'WEIGHT 'BOLD)
                                                OPTIONS))
                      (SETQ GRAPH (LAYOUTGRAPH GRAPHNODES (LIST FN)
                                           :FORMAT :FONT))
                      (if (OR (NULL :STREAM)
                              (WINDOWP :STREAM))
                          then [if (OR :ADVISE :SHAPE)
                                   then (LET (GRAPHWIDTH GRAPHHEIGHT)
                                              (with REGION (GRAPHREGION GRAPH)
                                                    (SETQ GRAPHWIDTH (WIDTHIFWINDOW (if (EQMEMB 'COUNT :ADVISE)
                                                                                       then (PLUS WIDTH
                                                                                                  (STRINGWIDTH
                                                                                                         "00000"
                                                                                                         :FONT))
                                                                                     else WIDTH)))
                                                    (SETQ GRAPHHEIGHT (HEIGHTIFWINDOW HEIGHT :ALLOWEDITFLG)))
                                              (if (WINDOWP :STREAM)
                                                  then (with REGION (WINDOWPROP :STREAM 'REGION)
                                                             (SHAPEW :STREAM (CREATEREGION LEFT BOTTOM GRAPHWIDTH
                                                                                   GRAPHHEIGHT)))
                                                else (SETQ :STREAM (CREATEW (GETBOXREGION GRAPHWIDTH GRAPHHEIGHT]
                               (SETQ :STREAM (SHOWGRAPH GRAPH :STREAM :LEFTBUTTONFN :MIDDLEBUTTONFN :TOPJUSTIFYFLG
                                                      :ALLOWEDITFLG :COPYBUTTONEVENTFN))
                               (WINDOWPROP :STREAM 'CLOSEFN (FUNCTION GRAPHCALLS.GRAPH.CLOSEFN))
                               (SETQ :STREAM (WINDOWPROP :STREAM 'DSP))
                        else (DISPLAYGRAPH GRAPH :STREAM))
                      [if :ADVISE
                          then (STREAMPROP :STREAM 'ADVISEDFNS (RESETFORM (CURSOR WAITINGCURSOR)
                                                                          (GRAPHCALLS.ADVISE GRAPHNODES FN
                                                                                 (MKLIST :ADVISE)
                                                                                 :STREAM]
                      (STREAMPROP :STREAM 'OPTIONS OPTIONS)
                      (RETURN :STREAM])
)

(DEFINEQ

(GRAPHCALLS1
  [LAMBDA (FN DEPTH LEAFFONT OPTIONS)                                 ; Edited 28-Jul-87 08:09 by cdl
    (DECLARE (USEDFREE GRAPHCALLS.SEEN))
    (PROG (GRAPHNODES GRAPHNODE)
           (push GRAPHCALLS.SEEN FN)
           [with GRAPHNODE [with GRAPHCALLS.OPTIONS OPTIONS (SETQ GRAPHNODE
                                                                (create GRAPHNODE
                                                                       NODEID _ FN
                                                                       NODELABEL _ (if :NAMEFN
                                                                                       then (APPLY* :NAMEFN FN OPTIONS
                                                                                                    )
                                                                                     elseif :PRIN2FLG
                                                                                       then (MKSTRING FN T)
                                                                                     else FN]
                 [with GRAPHCALLS.OPTIONS OPTIONS (if (NULL (SETQ TONODES (for SUBFN
                                                                                in (CAR (APPLY* :SEARCHFN FN OPTIONS))
                                                                                when (OR (NULL :FILTER)
                                                                                          (APPLY* :FILTER SUBFN))
                                                                                collect SUBFN)))
                                                        then (SETQ NODEFONT LEAFFONT)
                                                             (RETURN (CONS GRAPHNODE]
                 (if (ZEROP DEPTH)
                     then (SETQ TONODES NIL)
                   else (SETQ GRAPHNODES (for SUBFN in TONODES unless (FMEMB SUBFN GRAPHCALLS.SEEN)
                                             join (GRAPHCALLS1 SUBFN (SUB1 DEPTH)
                                                         LEAFFONT OPTIONS]
           (RETURN (CONS GRAPHNODE GRAPHNODES])

(GRAPHCALLS.SEARCH
  [LAMBDA (FN OPTIONS)                                                ; Edited 21-Jul-88 07:31 by cdl
    (if (FGETD FN)
        then (RESETLST
                  [if (with GRAPHCALLS.OPTIONS OPTIONS :SUBFNDEFFLG)
                      then (RESETSAVE (MOVD 'NILL '\SUBFNDEF)
                                   `(PUTD \SUBFNDEF ,(GETD '\SUBFNDEF]
                  (CALLS FN))])

(GRAPHCALLS.ADVISE
  [LAMBDA (GRAPHNODES ROOTID FLAGS STREAM)                            ; Edited 21-Jul-88 08:28 by cdl
    (DECLARE (GLOBALVARS UNSAFE.TO.MODIFY.FNS))
    (LET (ADVISED)
          (if (NOT (FMEMB ROOTID UNSAFE.TO.MODIFY.FNS))
              then (push ADVISED (GRAPHCALLS.ADVISE1 ROOTID (GETNODEFROMID ROOTID GRAPHNODES)
                                         FLAGS STREAM)))
          [bind PARENT for GRAPHNODE in GRAPHNODES do (SETQ PARENT (with GRAPHNODE GRAPHNODE
                                                                           (if (LISTP NODEID)
                                                                           then (CAR NODEID)
                                                                         else NODEID)))
                                                    (bind FN for TONODE in (with GRAPHNODE GRAPHNODE TONODES)
                                                       when (AND (SETQ FN (if (LISTP TONODE)
                                                                               then (CAR TONODE)
                                                                             else TONODE))
                                                                  (FGETD FN))
                                                       unless (FMEMB FN UNSAFE.TO.MODIFY.FNS)
                                                       do (push ADVISED (GRAPHCALLS.ADVISE1 (LIST FN :IN PARENT)
                                                                                (GETNODEFROMID TONODE GRAPHNODES)
                                                                                FLAGS STREAM]
         ADVISED])

(GRAPHCALLS.ADVISE1
  [LAMBDA (FN NODE FLAGS STREAM)                                  ; Edited 21-Jul-88 07:44 by cdl
    [ADVISE FN 'AROUND NIL `(PROG2 (GRAPHCALLS.INVERT.NODE ,(KWOTE NODE)
                                           ,STREAM
                                           ,(KWOTE FLAGS)
                                           'BEFORE)
                                    *
                                    (GRAPHCALLS.INVERT.NODE ,(KWOTE NODE)
                                           ,STREAM
                                           ,(KWOTE FLAGS)
                                           'AFTER]
    (UNMARKASCHANGED FN 'ADVICE)
    FN])

(GRAPHCALLS.BREAKIN
  [LAMBDA (NODE WINDOW FN)                                        ; Edited 24-Aug-88 08:53 by cdl
    (PROG [PARENT NODELST GRAPHNODEID (GRAPHNODES (fetch GRAPHNODES of (WINDOWPROP WINDOW 'GRAPH]
          (with GRAPHNODE NODE (if (NULL FROMNODES)
                                   then (RETURN))
                (SETQ NODELST (for FROMNODE in FROMNODES collect (GETNODEFROMID FROMNODE GRAPHNODES)))
                (if (CDR NODELST)
                    then (for NODE in NODELST do (FLIPNODE NODE WINDOW))
                         (GRAPHCALLS.PRINT "In the context of which node?")
                         (SETQ PARENT (READ/NODE NODELST WINDOW))
                         (for NODE in NODELST do (FLIPNODE NODE WINDOW))
                  else (SETQ PARENT (CAR NODELST)))
                (if PARENT
                    then (if (LISTP (SETQ GRAPHNODEID (fetch (GRAPHNODE NODEID) of PARENT)))
                             then (SETQ GRAPHNODEID (CAR GRAPHNODEID)))
                         (GRAPHCALLS.PRINT (APPLY* FN `(,NODELABEL :IN ,GRAPHNODEID])

(GRAPHCALLS.LEFT
  [LAMBDA (GRAPHNODE WINDOW)
    (DECLARE (SPECVARS GRAPHNODE WINDOW))                         (* cdl "15-Oct-85 10:06")
    (LET [FN (STREAM (WINDOWPROP WINDOW 'DSP]
         (DECLARE (SPECVARS FN STREAM))
         (if GRAPHNODE
             then (if (LISTP (SETQ FN (fetch (GRAPHNODE NODEID) of GRAPHNODE)))
                      then (SETQ FN (CAR FN)))
                  (MENU GRAPHCALLS.MENU)
           else (MENU GRAPHCALLS.BACKGROUND.MENU])

(GRAPHCALLS.MIDDLE
  [LAMBDA (GRAPHNODE WINDOW)
    (DECLARE (SPECVARS GRAPHNODE))                                (* cdl "15-Oct-85 10:07")
    (LET (FN)
         (DECLARE (SPECVARS FN))
         (if (AND GRAPHNODE GRAPHCALLSW.WINDOW)
             then (if (LISTP (SETQ FN (fetch (GRAPHNODE NODEID) of GRAPHNODE)))
                      then (SETQ FN (CAR FN)))
                  (MENU GRAPHCALLS.MIDDLE.MENU])

(GRAPHCALLS.COLLECT
  [LAMBDA (FN GRAPHNODES BACKFLG)                                 ; Edited 31-Mar-87 10:25 by cdl
    (DECLARE (USEDFREE GRAPHCALLS.COLLECTED))
    (LET [EXPANDCALLS CALLS (VARS (if (FGETD FN)
                                      then (VARS FN]
         [with GRAPHNODE (GETNODEFROMID FN GRAPHNODES)
               (SETQ EXPANDCALLS (for ID in (if BACKFLG
                                                then FROMNODES
                                              else TONODES)
                                    unless (FMEMB ID GRAPHCALLS.COLLECTED) collect (PROGN (push
                                                                                          GRAPHCALLS.COLLECTED
                                                                                          ID)
                                                                                    ID]
         [if EXPANDCALLS
             then (for ID in EXPANDCALLS when (FGETD ID) do (SETQ CALLS (GRAPHCALLS.COLLECT ID GRAPHNODES
                                                                               BACKFLG))
                                                            (with GRAPHCALLS.RECORD VARS
                                                                (SETQ FREEVARS (UNION FREEVARS
                                                                                      (fetch (GRAPHCALLS.RECORD
                                                                                             FREEVARS)
                                                                                         of CALLS)))
                                                                (SETQ GLOBALVARS (UNION GLOBALVARS
                                                                                        (fetch (GRAPHCALLS.RECORD
                                                                                               GLOBALVARS)
                                                                                           of CALLS]
         (replace (GRAPHCALLS.RECORD LOCALVARS) of VARS with NIL)
         VARS])

(GRAPHCALLS.INSPECT.FRAME
  [LAMBDA (FN WINDOW TREEFLG BACKFLG)                            (* cdl "10-Oct-85 17:41")
    (PROG (RECORD GRAPHCALLS.COLLECTED VARS)
          (DECLARE (SPECVARS GRAPHCALLS.COLLECTED))
          (if (AND TREEFLG (OR BACKFLG (FGETD FN))
                  (SETQ RECORD (GRAPHCALLS.COLLECT FN (fetch GRAPHNODES of (WINDOWPROP WINDOW 'GRAPH))
                                                    BACKFLG))
                  (in RECORD thereis LISTP))
              then (GRAPHCALLS.INSPECT RECORD WINDOW (CONCAT FN "'s " (if BACKFLG
                                                                           then "scope"
                                                                         else "tree")))
            elseif (AND (NOT TREEFLG)
                       (FGETD FN)
                       (in (SETQ VARS (VARS FN)) thereis LISTP))
              then (GRAPHCALLS.INSPECT VARS WINDOW FN)
            else (GRAPHCALLS.PRINT NIL "Nothing to INSPECT!"])

(GRAPHCALLS.INSPECT
  [LAMBDA (RECORD WINDOW LABEL)                                  ; Edited  1-Apr-87 08:21 by cdl
    (PROG [INSPECTW INSPECTWS REGION (LINEHEIGHT (FONTPROP DEFAULTFONT 'HEIGHT]
          (SETQ REGION (CREATEREGION NIL NIL (WIDTHIFWINDOW (with GRAPHCALLS.OPTIONS GRAPHCALLS.DEFAULT.OPTIONS
                                                                    :INSPECTWIDTH))
                              (HEIGHTIFWINDOW (TIMES (LENGTH (in RECORD thereis LISTP))
                                                     LINEHEIGHT)
                                     T)))
          (with REGION (with REGION REGION (GETBOXREGION WIDTH HEIGHT))
               (replace (REGION LEFT) of REGION with LEFT)
               (replace (REGION BOTTOM) of REGION with BOTTOM))
          [SETQ INSPECTWS (for FIELD in [CONSTANT (REVERSE (RECORDFIELDNAMES 'GRAPHCALLS.RECORD] as VALUE
                             in RECORD when VALUE
                             collect (PROG1 (SETQ INSPECTW (INSPECTW.CREATE
                                                          VALUE VALUE (FUNCTION GRAPHCALLS.FETCH)
                                                          (FUNCTION GRAPHCALLS.STORE)
                                                          NIL NIL NIL (if LABEL
                                                                          then (CONCAT FIELD " in " LABEL))
                                                          NIL
                                                          (create REGION
                                                                 HEIGHT _ (HEIGHTIFWINDOW (ITIMES (LENGTH VALUE)
                                                                                                  LINEHEIGHT)
                                                                                 T)
                                                                 BOTTOM _
                                                                 (if INSPECTW
                                                                     then (fetch (REGION TOP)
                                                                             of (WINDOWPROP INSPECTW 'REGION))
                                                                   else (fetch BOTTOM of REGION))
                                                             using REGION)))
                                        (WINDOWPROP INSPECTW 'CLOSEFN (FUNCTION GRAPHCALLS.CLOSE))
                                        (WINDOWPROP INSPECTW 'GRAPHW WINDOW))]
          (if INSPECTWS
              then (WINDOWADDPROP WINDOW 'INSPECTWS INSPECTWS])

(GRAPHCALLS.INVERT.NODE
  [LAMBDA (NODE STREAM FLAGS WHEN)                               ; Edited 28-Jul-87 15:48 by cdl
    (if (FMEMB 'INVERT FLAGS)
        then (FLIPNODE NODE STREAM))
    (SELECTQ WHEN
         (BEFORE (BLOCK (with GRAPHCALLS.OPTIONS GRAPHCALLS.DEFAULT.OPTIONS :DELAY)))
         (AFTER [if (FMEMB 'COUNT FLAGS)
                   then (with GRAPHNODE NODE (MOVETO (PLUS (fetch XCOORD of NODEPOSITION)
                                                           (QUOTIENT NODEWIDTH 2))
                                                    (DIFFERENCE (PLUS (fetch YCOORD of NODEPOSITION)
                                                                      (FONTPROP NODEFONT 'DESCENT)
                                                                     1)
                                                           (QUOTIENT NODEHEIGHT 2))
                                                    STREAM)
                              (DSPFONT NODEFONT STREAM)
                              (printout STREAM %, (if (FIXP NODEBORDER)
                                                     then (add NODEBORDER 1)
                                                   else (SETQ NODEBORDER 1])
         (SHOULDNT])

(GRAPHCALLS.FETCH
  [LAMBDA (OBJECT PROPERTY)                                      (* cdl "21-Feb-84 14:17")
    (EVALV PROPERTY])

(GRAPHCALLS.STORE
  [LAMBDA (OBJECT PROPERTY NEWVALUE)                             (* cdl "28-Feb-84 10:09")
    (SET PROPERTY NEWVALUE])

(GRAPHCALLS.PRINT
  [LAMBDA (EXP ERROR)                                            ; Edited 28-Jul-87 16:07 by cdl
    (DECLARE (GLOBALVARS PROMPTWINDOW))
    (CLRPROMPT)
    (if EXP
        then (CENTERPRINTINREGION EXP NIL PROMPTWINDOW)
      elseif ERROR
        then (RINGBELLS)
             (CENTERPRINTINREGION ERROR NIL PROMPTWINDOW])

(GRAPHCALLS.CLOSE
  [LAMBDA (WINDOW)                                               ; Edited 31-Mar-87 10:15 by cdl
    (PROG (INSPECTWS (GRAPHW (WINDOWPROP WINDOW 'GRAPHW NIL)))
          (SETQ INSPECTWS (for WINDOWLST in (WINDOWPROP GRAPHW 'INSPECTWS) thereis (FMEMB WINDOW WINDOWLST)))
          (for INSPECTW in INSPECTWS when (AND (OPENWP INSPECTW)
                                               (NEQ INSPECTW WINDOW))
             do (WINDOWPROP INSPECTW 'CLOSEFN NIL)
                (WINDOWPROP INSPECTW 'GRAPHW NIL)
                (CLOSEW INSPECTW))
          (WINDOWDELPROP GRAPHW 'INSPECTWS INSPECTWS])

(GRAPHCALLS.GRAPH.CLOSEFN
  [LAMBDA (WINDOW)                                               (* cdl "27-Jun-85 15:13")
    (for INSPECTWS in (WINDOWPROP WINDOW 'INSPECTWS NIL) do (for INSPECTW in INSPECTWS when (OPENWP INSPECTW)
                                                              do (WINDOWPROP INSPECTW 'CLOSEFN NIL)
                                                                 (WINDOWPROP INSPECTW 'GRAPHW NIL)
                                                                 (CLOSEW INSPECTW)))
    (LET ((FNS (STREAMPROP (WINDOWPROP WINDOW 'DSP)
                      'ADVISEDFNS NIL)))
         (if FNS
             then (APPLY (FUNCTION UNADVISE)
                         FNS])

(NO\
  [LAMBDA (FN)                                                   (* cdl " 6-Mar-84 14:47")
    (NEQ (NTHCHARCODE FN 1)
         (CHARCODE \])

(GRAPHCALLS.INSPECTCODE
  [LAMBDA (FN)                                                   ; Edited 17-Sep-87 08:52 by cdl
    (DECLARE (GLOBALVARS SCREENHEIGHT SCROLLBARWIDTH))
    (LET ((TITLE (CONCAT FN " Code Window")))
         (if (NOT (WINDOWP GRAPHCALLS.INSPECTCODE.WINDOW))
             then (SETQ GRAPHCALLS.INSPECTCODE.WINDOW (CREATEW (GETBOXREGION (with GRAPHCALLS.OPTIONS
                                                                                  GRAPHCALLS.DEFAULT.OPTIONS
                                                                                  :INSPECTCODEWIDTH)
                                                                      (DIFFERENCE SCREENHEIGHT SCROLLBARWIDTH))
                                                             TITLE))
           else (WINDOWPROP GRAPHCALLS.INSPECTCODE.WINDOW 'TITLE TITLE)
                (DSPRESET GRAPHCALLS.INSPECTCODE.WINDOW))
         (OR (INSPECTCODE FN GRAPHCALLS.INSPECTCODE.WINDOW)
             FN])

(GRAPHCALLS.WHEREIS
  [LAMBDA (FN OPTIONS)                                           ; Edited 24-Aug-88 08:57 by cdl
    (LET ((FILES (for TYPE in '(FNS FUNCTIONS) join (WHEREIS FN TYPE T)))
          (LABEL (if (with GRAPHCALLS.OPTIONS OPTIONS :PRIN2FLG)
                     then (MKSTRING FN T)
                   else FN)))
         (if FILES
             then (CONS LABEL FILES)
           else LABEL])

(GRAPHCALLS.ARGLIST
  [LAMBDA (FN OPTIONS)                                           ; Edited  1-Apr-87 08:58 by cdl
    (LET ((LABEL (if (with GRAPHCALLS.OPTIONS OPTIONS :PRIN2FLG)
                     then (MKSTRING FN T)
                   else FN)))
         (if (FGETD FN)
             then (CONS LABEL (SMARTARGLIST FN))
           else LABEL])
```

```
)

(RPAQ? GRAPHCALLS.INSPECTCODE.WINDOW NIL)

(RPAQ? GRAPHCALLS.DEFAULT.OPTIONS '(:DELAY 500 :SUBFNDEFFLG T :SEARCHFN GRAPHCALLS.SEARCH :DEPTH 2
                                             :LEFTBUTTONFN GRAPHCALLS.LEFT :MIDDLEBUTTONFN GRAPHCALLS.MIDDLE
                                             :INSPECTWIDTH 250 :INSPECTCODEWIDTH 400 :FONT (GACHA 8)
                                             :FORMAT
                                             (HORIZONTAL COMPACT REVERSE/DAUGHTERS)))

(PUTPROPS GRAPHCALLS ARGNAMES (NIL (FN &KEY :ADVISE :ALLOWEDITFLG :COPYBUTTONEVENTFN :DELAY :DEPTH :FILTER
                                           :FONT :FORMAT :INSPECTCODEWIDTH :INSPECTWIDTH :LEFTBUTTONFN
                                           :MIDDLEBUTTONFN :NAMEFN :PRIN2FLG :SEARCHFN :SHAPE :STREAM
                                           :SUBFNDEFFLG :TOPJUSTIFYFLG)))

(RPAQQ GRAPHCALLS.MENUS (GRAPHCALLS.BACKGROUND.MENU GRAPHCALLS.MENU GRAPHCALLS.MIDDLE.MENU
                                  GRAPHCALLS.SOURCE.MENU))

(PUTPROPS GRAPHCALLS.BACKGROUND.MENU MENU [create MENU ITEMS _
                                                       '(("UNBREAK" (UNBREAK)
                                                                "UnBreak everything.")
                                                         ("RESET" (for GRAPHNODE in (fetch (GRAPH GRAPHNODES)
                                                                                            of
                                                                                            (WINDOWPROP
                                                                                             WINDOW
                                                                                             'GRAPH))
                                                                       when
                                                                       (with GRAPHNODE GRAPHNODE (FIXP NODEBORDER
                                                                                                       ))
                                                                       do
                                                                       (with GRAPHNODE GRAPHNODE (SETQ NODEBORDER
                                                                                                       NIL))
                                                                       finally
                                                                       (REDISPLAYW WINDOW])

(PUTPROPS GRAPHCALLS.MENU MENU
          (create MENU ITEMS _ '[("?=" (PROG (ARGS)
                                              (GRAPHCALLS.PRINT [COND ((SETQ ARGS (NLSETQ (SMARTARGLIST FN T)))
                                                                        `(,FN ,@(CAR ARGS]
                                                     "Args not availiable!"))
                                        "The function's argument list")
                                  ("HELP" (GRAPHCALLS.PRINT (NLSETQ (PROGN (IRM.LOOKUP FN)
                                                                            FN))
                                                 "Help not available!")
                                         "HelpSys information")
                                  ("FNTYP" (GRAPHCALLS.PRINT (FNTYP FN)
                                                  "Fn's type not found")
                                         "Get the FNTYP of the function")
                                  ("WHERE" (GRAPHCALLS.PRINT (WHEREIS FN NIL T)
                                                  "File not found!")
                                         "Do a WHEREIS on function")
                                  ("EDIT" (GRAPHCALLS.PRINT (NLSETQ (PROG1 (EDITDEF FN 'FNS)
                                                                            (TOTOPW WINDOW)))
                                                 "Nothing to EDIT!")
                                         "Edit the function")
                                  ("TYPEIN" (BKSYSBUF FN T)
                                         "BKSYSBUF the function name")
                                  ("BREAK" (GRAPHCALLS.PRINT (APPLY* (FUNCTION BREAK)
                                                                    FN))
                                         "Break this function"
                                         (SUBITEMS ("BREAKIN" (GRAPHCALLS.BREAKIN GRAPHNODE WINDOW
                                                                     (FUNCTION BREAK))
                                                          "Break this fn in another fn")
                                                ("UNBREAKIN" (GRAPHCALLS.BREAKIN GRAPHNODE WINDOW
                                                                    (FUNCTION UNBREAK))
                                                       "UnBreak this fn in another fn")
                                                ("UNBREAK" (GRAPHCALLS.PRINT (APPLY* (FUNCTION UNBREAK)
                                                                                    FN))
                                                       "UnBreak this function")
                                                ("TRACE" (GRAPHCALLS.PRINT (APPLY* (FUNCTION TRACE)
                                                                                  FN))
                                                       "Trace this function")
                                                ("TRACEIN" (GRAPHCALLS.BREAKIN GRAPHNODE WINDOW
                                                                  (FUNCTION TRACE))
                                                       "Trace this fn in another fn")))
                                  ("CCODE" (GRAPHCALLS.PRINT (COND ((CCODEP FN)
                                                                     (GRAPHCALLS.INSPECTCODE FN)))
                                                  "Not compiled code!")
                                         "Inspect this function's ccode")
                                  ("GRAPH" (GRAPHCALLS.PRINT [APPLY (FUNCTION GRAPHCALLS)
                                                                    `(,FN :STREAM NIL :DEPTH
                                                                          ,(with GRAPHCALLS.OPTIONS
                                                                                 GRAPHCALLS.DEFAULT.OPTIONS
                                                                                 :DEPTH)
                                                                          ,@(APPEND (STREAMPROP STREAM
                                                                                           'OPTIONS]
                                                  "Nothing to graph!")
                                         "Graph this function's calls")
                                  ("FRAME" (GRAPHCALLS.INSPECT.FRAME FN WINDOW)
                                         "Inspect this function's vars"
                                         (SUBITEMS (">FRAME" (GRAPHCALLS.INSPECT.FRAME FN WINDOW T)
                                                          "Inspect this sub-graph's freevars")
                                                ("<FRAME" (GRAPHCALLS.INSPECT.FRAME FN WINDOW T T)
                                                       "Inspect this fn's scope"]
                  CENTERFLG _ T))

 (PUTPROPS GRAPHCALLS.MIDDLE.MENU MENU
           (create MENU ITEMS _ '((EXCLUDE [PROG [(ALIST (WINDOWPROP GRAPHCALLSW.WINDOW 'EXCLUDEFNS))
                                                  (FILE (OR (CAR (WHEREIS FN 'FNS FILELST))
                                                            'SYSTEM]
                                                 (if ALIST then (PUTASSOC FILE (CONS FN (CDR (ASSOC FILE ALIST)))
                                                                       ALIST)
                                                     else
                                                     (WINDOWADDPROP GRAPHCALLSW.WINDOW 'EXCLUDEFNS
                                                            (LIST FILE FN]
                                         "Exclude this function from future graphs"))
                  CENTERFLG _ T))

 (PUTPROPS GRAPHCALLS.SOURCE.MENU MENU (create MENU ITEMS _ '(FILES FNS)
                                                 CENTERFLG _ T))

 (RPAQQ GRAPHCALLS.MENUS (GRAPHCALLS.BACKGROUND.MENU GRAPHCALLS.MENU GRAPHCALLS.MIDDLE.MENU
                                  GRAPHCALLS.SOURCE.MENU))

 (RPAQ? GRAPHCALLS.BACKGROUND.MENU NIL)

 (RPAQ? GRAPHCALLS.MENU NIL)

 (RPAQ? GRAPHCALLS.MIDDLE.MENU NIL)

 (RPAQ? GRAPHCALLS.SOURCE.MENU NIL)

 (RPAQQ GRAPHCALLS.MENUS (GRAPHCALLS.BACKGROUND.MENU GRAPHCALLS.MENU GRAPHCALLS.MIDDLE.MENU
                                  GRAPHCALLS.SOURCE.MENU))

 (DECLARE%: DOEVAL@COMPILE DONTCOPY

 (GLOBALVARS GRAPHCALLS.BACKGROUND.MENU GRAPHCALLS.MENU GRAPHCALLS.MIDDLE.MENU GRAPHCALLS.SOURCE.MENU)
 )

 (DECLARE%: DOEVAL@COMPILE DONTCOPY

 (GLOBALVARS GRAPHCALLS.DEFAULT.OPTIONS GRAPHCALLS.INSPECTCODE.WINDOW)
 )

 (DECLARE%: EVAL@COMPILE

 (RECORD GRAPHCALLS.RECORD (LOCALVARS FREEVARS GLOBALVARS))

 (PROPRECORD GRAPHCALLS.OPTIONS
        (:DELAY :SEARCHFN :DEPTH :FORMAT :SHAPE :FILTER :STREAM :ADVISE :LEFTBUTTONFN :MIDDLEBUTTONFN
               :SUBFNDEFFLG :NAMEFN :FONT :TOPJUSTIFYFLG :ALLOWEDITFLG :COPYBUTTONEVENTFN :PRIN2FLG :INSPECTWIDTH
               :INSPECTCODEWIDTH))
 )

 (DECLARE%: DOEVAL@COMPILE DONTCOPY

 (GLOBALVARS DEFAULT.GRAPH.NODEBORDER DEFAULT.GRAPH.NODEFONT DEFAULT.GRAPH.NODELABELSHADE)
 )

 (DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY

 (BLOCK%: GRAPHCALLS (SPECVARS GRAPHCALLS.SEEN)
        (ENTRIES GRAPHCALLS)
        GRAPHCALLS GRAPHCALLS1 GRAPHCALLS.ADVISE GRAPHCALLS.ADVISE1)

 (BLOCK%: GRAPHCALLS.INSPECT.FRAME (SPECVARS GRAPHCALLS.COLLECTED)
        (ENTRIES GRAPHCALLS.INSPECT.FRAME)
        GRAPHCALLS.INSPECT GRAPHCALLS.INSPECT.FRAME GRAPHCALLS.COLLECT)
 )

 (FILESLOAD (SYSLOAD FROM LISPUSERS)
        MSANALYZE GRAPHER)

 [for MENU in GRAPHCALLS.MENUS do (SET MENU (EVAL (GETPROP MENU 'MENU]


           (* * GRAPHCALLS Command Window)

 (DEFINEQ

 (GRAPHCALLSW
   [LAMBDA (REBUILD?)                                              ; Edited 24-Mar-87 12:26 by cdl
     (PROG (MENULST DEFAULT PROMPTW)
           (if (AND (WINDOWP GRAPHCALLSW.WINDOW)
                   (NOT REBUILD?))
               then (with REGION (with REGION (WINDOWPROP GRAPHCALLSW.WINDOW 'REGION)
                                     (GETBOXREGION WIDTH HEIGHT))
                         (MOVEW GRAPHCALLSW.WINDOW LEFT BOTTOM))
                    (RETURN (TOTOPW GRAPHCALLSW.WINDOW))
             elseif (OPENWP GRAPHCALLSW.WINDOW)
               then (CLOSEW GRAPHCALLSW.WINDOW))
           [SETQ MENULST (for MENU in GRAPHCALLSW.MENUS collect (EVAL (GETPROP MENU 'MENU]
           (SETQ GRAPHCALLSW.WINDOW (CREATEW (GETBOXREGION [WIDTHIFWINDOW (ADD1 (for MENU in MENULST
                                                                                   sum (SUB1 (fetch IMAGEWIDTH
                                                                                                 of MENU]
                                                    (HEIGHTIFWINDOW (fetch (MENU IMAGEHEIGHT)
                                                                       of (for MENU in MENULST
                                                                            largest (fetch (MENU IMAGEHEIGHT)
                                                                                      of MENU)))
                                                           T))
                                           "GraphCalls Command Window"))
           [bind (HEIGHT _ (WINDOWPROP GRAPHCALLSW.WINDOW 'HEIGHT))
                 (POSITION _ (create POSITION
                                    XCOORD _ 0))
              for MENU in MENULST do (with MENU MENU (with POSITION POSITION (SETQ YCOORD (IDIFFERENCE HEIGHT
                                                                                                IMAGEHEIGHT))
                                                         (ADDMENU MENU GRAPHCALLSW.WINDOW POSITION)
                                                         (if (SETQ DEFAULT (GETMENUPROP MENU 'VALUE))
                                                            then (DOSELECTEDITEM MENU DEFAULT))
                                                         (add XCOORD (SUB1 IMAGEWIDTH]
           [WINDOWPROP GRAPHCALLSW.WINDOW 'REPAINTFN (FUNCTION (LAMBDA (WINDOW)
                                                                 (MENUREPAINTFN GRAPHCALLSW.WINDOW)
                                                                 (GRAPHCALLSW.CLEAR)
                                                                 (GRAPHCALLSW.PRINTFN (WINDOWPROP
                                                                                               GRAPHCALLSW.WINDOW
                                                                                               'FUNCTION]
           [WINDOWPROP GRAPHCALLSW.WINDOW 'AFTERMOVEFN (FUNCTION (LAMBDA (WINDOW)
                                                                   (with REGION (WINDOWPROP WINDOW 'REGION)
                                                                        (MOVEW (WINDOWPROP WINDOW 'PROMPTWINDOW)
                                                                               LEFT PTOP]
           (WINDOWPROP GRAPHCALLSW.WINDOW 'PROMPTWINDOW (SETQ PROMPTW
                                                         (CREATEW [with REGION (WINDOWPROP GRAPHCALLSW.WINDOW
                                                                                     'REGION)
                                                                        (CREATEREGION LEFT PTOP WIDTH
                                                                               (HEIGHTIFWINDOW (FONTPROP
                                                                                                (DSPFONT NIL
                                                                                                GRAPHCALLSW.WINDOW
                                                                                                      )
                                                                                               'HEIGHT]
                                                                NIL NIL T)))
           (WINDOWPROP PROMPTW 'PAGEFULLFN (FUNCTION NILL))
           (RETURN GRAPHCALLSW.WINDOW])
 )

 (DEFINEQ

 (GRAPHCALLSW.CLEAR
   [LAMBDA NIL                                                       ; Edited  1-Apr-87 08:21 by cdl
     (LET [(MENUS (WINDOWPROP GRAPHCALLSW.WINDOW 'MENU]
          (for MENU in MENUS do (with MENU MENU (for ITEM in SHADEDITEMS do (RPLACD ITEM 0)))
                               (PUTMENUPROP MENU 'VALUE NIL))
          (MENUREPAINTFN GRAPHCALLSW.WINDOW)
          (with GRAPHCALLSW.MENUS.RECORD MENUS (DOSELECTEDITEM DELAYMENU 5)
               (DOSELECTEDITEM DEPTHMENU (with GRAPHCALLS.OPTIONS GRAPHCALLS.DEFAULT.OPTIONS :DEPTH)))
          (for PROP in '(INCLUDESET INCLUDEFNS INCLUDEFILES EXCLUDESET EXCLUDEFNS EXCLUDEFILES)
             do (WINDOWPROP GRAPHCALLSW.WINDOW PROP NIL])

 (GRAPHCALLSW.DOIT
   [LAMBDA NIL                                                       ; Edited 21-Jul-88 07:50 by cdl
     (PROG [FLAGS FORMAT NAMEFN ADVISEMODE (FN (WINDOWPROP GRAPHCALLSW.WINDOW 'FUNCTION]
           (WINDOWPROP GRAPHCALLSW.WINDOW 'INCLUDESET (in (WINDOWPROP GRAPHCALLSW.WINDOW 'INCLUDEFNS) join CDR))
           (WINDOWPROP GRAPHCALLSW.WINDOW 'EXCLUDESET (in (WINDOWPROP GRAPHCALLSW.WINDOW 'EXCLUDEFNS) join CDR))
           [if FN
               then (with GRAPHCALLSW.MENUS.RECORD (WINDOWPROP GRAPHCALLSW.WINDOW 'MENU)
                         [SETQ FLAGS (for VALUE in (GETMENUPROP FLAGSMENU 'VALUE) join (COPY (CADR VALUE]
                         (SETQ FORMAT (in (GETMENUPROP FORMATMENU 'VALUE) collect CADR))
                         (if (SETQ NAMEFN (ASSOC :NAMEFN FORMAT))
                             then (SETQ FORMAT (DREMOVE NAMEFN FORMAT))
                                  (SETQ NAMEFN (CADR NAMEFN)))
                         (if FORMAT
                             then (push FORMAT 'REVERSE/DAUGHTERS))
                         (if (LISTGET FLAGS :ADVISE)
                             then (LISTPUT FLAGS :ADVISE (for FLAG on FLAGS when (EQ (CAR FLAG)
                                                                                    :ADVISE)
                                                           collect (CADR FLAG)))
                                  (if [AND (for ITEM in (GETMENUPROP FILTERMENU 'VALUE)
                                              never (EQ (CADR ITEM)
                                                       'WHEREIS))
```

```
                                        (NOT (MOUSECONFIRM "Advising without a WHEREIS filter advises system
                                                    code, confirm to continue." (CONSTANT null]
                                    then (RETURN)))
                        (if [APPLY (FUNCTION GRAPHCALLS)
                                  `(,FN :FILTER GRAPHCALLSW.FILTER :DEPTH ,(GETMENUPROP DEPTHMENU 'VALUE)
                                        :FORMAT
                                        ,FORMAT :NAMEFN ,NAMEFN ,@FLAGS]
                            then (RETURN]
          (GRAPHCALLS.PRINT NIL "Can't graph function"])

(GRAPHCALLSW.FILTER
  [LAMBDA (FN)                                                  (* cdl "15-Oct-85 14:54")
    (PROG [(INCLUDESET (WINDOWPROP GRAPHCALLSW.WINDOW 'INCLUDESET))
           (EXCLUDESET (WINDOWPROP GRAPHCALLSW.WINDOW 'EXCLUDESET))
           (EXCLUDEFILES (WINDOWPROP GRAPHCALLSW.WINDOW 'EXCLUDEFILES))
           (INCLUDEFILES (WINDOWPROP GRAPHCALLSW.WINDOW 'INCLUDEFILES]
          (if (NULL (for FILTER in (GETMENUPROP (with GRAPHCALLSW.MENUS.RECORD (WINDOWPROP GRAPHCALLSW.WINDOW
                                                                                  'MENU)
                                                   FILTERMENU)
                                       'VALUE)
                      always (APPLY* (CADR FILTER)
                                     FN)))
              then (RETURN)
            elseif (AND EXCLUDESET (FMEMB FN EXCLUDESET))
              then (RETURN)
            elseif (AND INCLUDESET (FMEMB FN INCLUDESET))
              then (RETURN FN)
            elseif (AND EXCLUDEFILES (WHEREIS FN 'FNS EXCLUDEFILES))
              then (RETURN)
            elseif INCLUDEFILES
              then (RETURN (WHEREIS FN 'FNS INCLUDEFILES))
            else (RETURN FN])

(GRAPHCALLSW.INCLUDE
  [LAMBDA (INFLG)                                               ; Edited 31-Mar-87 13:33 by cdl
    (DECLARE (GLOBALVARS FILELST))
    (PROG (FILES FNS SOURCE ALIST LABEL)
          (SETQ LABEL (if INFLG
                          then "Include "
                        else "Exclude "))
          (if (NULL (SETQ SOURCE (MENU GRAPHCALLS.SOURCE.MENU)))
              then (RETURN)
            elseif (EQ SOURCE 'FILES)
              then [SETQ FILES (WINDOWPROP GRAPHCALLSW.WINDOW (if INFLG
                                                                  then 'INCLUDEFILES
                                                                else 'EXCLUDEFILES]
                   (SETQ FILES (MMENU (create MENU
                                             ITEMS _ (APPEND FILELST '(DONE))
                                             TITLE _ (CONCAT LABEL "Files") smashing GRAPHCALLSW.SCRATCHMENU)
                                     FILES))
                   (WINDOWPROP GRAPHCALLSW.WINDOW (if INFLG
                                                      then 'INCLUDEFILES
                                                    else 'EXCLUDEFILES)
                         FILES)
            elseif (AND (SETQ FILES (MENU (create MENU
                                               ITEMS _ FILELST
                                               TITLE _ 'FILELST smashing GRAPHCALLSW.SCRATCHMENU)))
                      (SETQ FNS (FILEFNSLST FILES)))
              then [SETQ ALIST (WINDOWPROP GRAPHCALLSW.WINDOW (if INFLG
                                                                  then 'INCLUDEFNS
                                                                else 'EXCLUDEFNS]
                   [SETQ FNS (MMENU (create MENU
                                           ITEMS _ (APPEND (FILEFNSLST FILES)
                                                          '(DONE))
                                           TITLE _ (CONCAT LABEL "from " FILES) smashing
                                                                                      GRAPHCALLSW.SCRATCHMENU)
                                    (CDR (ASSOC FILES ALIST]
                   (if ALIST
                       then (PUTASSOC FILES FNS ALIST)
                     else (WINDOWPROP GRAPHCALLSW.WINDOW (if INFLG
                                                             then 'INCLUDEFNS
                                                           else 'EXCLUDEFNS)
                                (LIST (CONS FILES FNS])

(GRAPHCALLSW.PRINTFN
  [LAMBDA (FN)                                                  ; Edited 21-Jul-88 07:53 by cdl
    (if FN
        then (WINDOWPROP GRAPHCALLSW.WINDOW 'FUNCTION FN)
             (with GRAPHCALLSW.MENUS.RECORD (WINDOWPROP GRAPHCALLSW.WINDOW 'MENU)
                  (LET ((WIDTH (with MENU DEPTHMENU MENUREGIONLEFT))
                        (HEIGHT (with MENU FILTERMENU MENUREGIONBOTTOM)))
                       (BLTSHADE (CONSTANT WHITESHADE)
                              GRAPHCALLSW.WINDOW 0 0 WIDTH HEIGHT)
                       (CENTERPRINTINAREA FN 0 0 WIDTH HEIGHT GRAPHCALLSW.WINDOW (with GRAPHCALLSW.MENUS.RECORD
                                                                                        (WINDOWPROP
                                                                                               GRAPHCALLSW.WINDOW
                                                                                               ’MENU)
                                                                                        T])

)

(RPAQ? GRAPHCALLSW.WINDOW NIL)

(RPAQ? GRAPHCALLSW.SCRATCHMENU (create MENU))

(RPAQQ GRAPHCALLSW.MENUS (GRAPHCALLSW.COMMAND.MENU GRAPHCALLSW.FILTER.MENU GRAPHCALLSW.FLAGS.MENU
                                   GRAPHCALLSW.FORMAT.MENU GRAPHCALLSW.DEPTH.MENU GRAPHCALLSW.DELAY.MENU))

(PUTPROPS GRAPHCALLSW.COMMAND.MENU MENU (create MENU ITEMS _
                                                     `((,(MMENU.BOLDITEM "Function" MENUFONT)
                                                        (PROG [(WINDOW (WINDOWPROP GRAPHCALLSW.WINDOW
                                                                              ’PROMPTWINDOW]
                                                              (GRAPHCALLSW.PRINTFN (CAR (PROCESS.READ WINDOW
                                                                                               "> " T)))
                                                              (CLOSEW WINDOW))
                                                        "Enter a new function to be graphed, prompts for input.")
                                                       (Include (GRAPHCALLSW.INCLUDE T)
                                                              "Specify which functions (by file or function) to
                                                              include (overide EXCLUDE).")
                                                       (Exclude (GRAPHCALLSW.INCLUDE NIL)
                                                              "Specify which functions (by file or function) to
                                                              exclude from the graph.")
                                                       (Clear (GRAPHCALLSW.CLEAR)
                                                              "Clear the current settings on the command window
                                                              to the defaults.")
                                                       (,(MMENU.BOLDITEM "Graph" MENUFONT)
                                                        (GRAPHCALLSW.DOIT)
                                                        "Graph the function with the selected settings."))
                                                     TITLE _ "Command" CENTERFLG _ T))

(PUTPROPS GRAPHCALLSW.FILTER.MENU MENU (create MENU ITEMS _ ’((WhereIs WHEREIS "Only graph functions that
                                                                         WHEREIS can locate.")
                                                                  (FGetD FGETD "Only graph functions that are
                                                                         defined.")
                                                                  (ExprP EXPRP "Only graph functions that are not
                                                                         compiled.")
                                                                  (CCodeP CCODEP "Only graph functions that are
                                                                         compiled.")
                                                                  (No\ NO\ "Only graph functions that do not have
                                                                       an initial slash in their name."))
                                                  TITLE _ "Filters" CENTERFLG _ T WHENSELECTEDFN _
                                                  (FUNCTION MMENU.SELECTEDFN)))

(PUTPROPS GRAPHCALLSW.FLAGS.MENU MENU (create MENU ITEMS _ ’((Invert (:ADVISE INVERT)
                                                                         "ADVISE the graphed functions to invert
                                                                         their node when called.")
                                                                  (Count (:ADVISE COUNT)
                                                                         "ADVISE the graphed functions to keep a
                                                                         count of calls after their node.")
                                                                  (Shape (:SHAPE T)
                                                                         "Shape the graph window to fit the
                                                                         graph.")
                                                                  (Edit (:ALLOWEDITFLG T)
                                                                        "Make the graph editable by passing the
                                                                        ALLOWEDITFLG to SHOWGRAPH.")
                                                                  (Prin2 (:PRIN2FLG T)
                                                                         "Display the package names."))
                                                 TITLE _ "Flags" CENTERFLG _ T WHENSELECTEDFN _
                                                 (FUNCTION MMENU.SELECTEDFN)))

(PUTPROPS GRAPHCALLSW.FORMAT.MENU MENU (create MENU ITEMS _ ’((Lattice LATTICE "Specify the LATTICE format
                                                                          in LAYOUTGRAPH.")
                                                                   (Reverse REVERSE "Specify the REVERSE format
                                                                          in LAYOUTGRAPH.")
                                                                   (Vertical VERTICAL "Specify the VERTICAL
                                                                          format in LAYOUTGRAPH.")
                                                                   (ArgList (:NAMEFN GRAPHCALLS.ARGLIST)
                                                                          "Use the function and its arguments as
                                                                          the node label.")
                                                                   (WhereIs (:NAMEFN GRAPHCALLS.WHEREIS)
                                                                          "Use the function and the file(s) where
                                                                          it is found as the node label."))
                                                   TITLE _ "Format" CENTERFLG _ T WHENSELECTEDFN _
                                                   (FUNCTION MMENU.SELECTEDFN)))

(PUTPROPS GRAPHCALLSW.DEPTH.MENU MENU (create MENU ITEMS _
                                                 ’(0 1 2 3 4 5 6 7 8 9 10)
                                                 TITLE _ "Depth" WHENSELECTEDFN _ (FUNCTION MMENU.MARKITEM)
                                                 MENUUSERDATA _ `(VALUE ,(with GRAPHCALLS.OPTIONS
                                                                              GRAPHCALLS.DEFAULT.OPTIONS :DEPTH
                                                                              ))
                                                 CENTERFLG _ T))

(PUTPROPS GRAPHCALLSW.DELAY.MENU MENU (create MENU ITEMS _
                                                 '(0 1 2 3 4 5 6 7 8 9 10)
                                                 TITLE _ "Delay" WHENSELECTEDFN _ [FUNCTION
                                                                                   (LAMBDA (ITEM MENU KEY)
                                                                                          (MMENU.MARKITEM ITEM
                                                                                                 MENU KEY)
                                                                                          (LISTPUT
                                                                                       GRAPHCALLS.DEFAULT.OPTIONS
                                                                                                 :DELAY
                                                                                                 (TIMES 100 ITEM
                                                                                                        ]
                                                 MENUUSERDATA _ '(VALUE 5)
                                                 CENTERFLG _ T))

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS GRAPHCALLSW.WINDOW GRAPHCALLSW.SCRATCHMENU GRAPHCALLSW.MENUS)
)

(DECLARE%: DONTCOPY

(DECLARE%: EVAL@COMPILE

(RECORD GRAPHCALLSW.MENUS.RECORD (COMMANDMENU FILTERMENU FLAGSMENU FORMATMENU DEPTHMENU DELAYMENU))
)
)

(ADDTOVAR UNSAFE.TO.MODIFY.FNS ERROR ERRORX RAID RECLAIM \ALLOCBLOCK \MOVEBYTES \MP.ERROR \STOP.DRIBBLE?)

(ADDTOVAR BackgroundMenuCommands (GraphCalls '(GRAPHCALLSW)
                                          "Open the GraphCalls Command Window"))

(RPAQQ BackgroundMenu NIL)


          (* * Multiple Selection Menus)

(DEFINEQ

(MMENU
  [LAMBDA (MENU PRESELECT)                                       ; Edited 31-Mar-87 14:38 by cdl
    (LET [(EVENT (GETMENUPROP MENU 'EVENT]
         [if (NULL EVENT)
             then (PUTMENUPROP MENU 'EVENT (SETQ EVENT (CREATE.EVENT 'MULTIMENU]
         (with MENU MENU (SETQ WHENSELECTEDFN (FUNCTION MMENU.SELECTEDFN)))
         (ADDMENU MENU)
         (for ITEM in PRESELECT do (DOSELECTEDITEM MENU ITEM))
         (AWAIT.EVENT EVENT)
         (PROG1 (GETMENUPROP MENU 'VALUE)
             (PUTMENUPROP MENU 'VALUE NIL))])

(MMENU.SELECTEDFN
  [LAMBDA (ITEM MENU KEY)                                        ; Edited 21-Jul-88 07:51 by cdl
    (LET [(VALUE (GETMENUPROP MENU 'VALUE]
         (SELECTQ ITEM
              (DONE (CLRPROMPT)
                    (SETQ VALUE (for ITEM in VALUE collect (DEFAULTWHENSELECTEDFN ITEM MENU)))
                    (DELETEMENU MENU T)
                    (NOTIFY.EVENT (GETMENUPROP MENU 'EVENT)))
              (CLEAR (for ITEM in VALUE do (SHADEITEM ITEM MENU (CONSTANT WHITESHADE)))
                     (SETQ VALUE NIL))
              (if (MEMB ITEM VALUE)
                  then (SHADEITEM ITEM MENU (CONSTANT WHITESHADE))
                        (SETQ VALUE (DREMOVE ITEM VALUE))
                else (SHADEITEM ITEM MENU (CONSTANT BLACKSHADE))
                     (push VALUE ITEM)))
         (PUTMENUPROP MENU 'VALUE VALUE])

(MMENU.MARKITEM
  [LAMBDA (ITEM MENU KEY)                                        ; Edited 21-Jul-88 07:52 by cdl
    [LET [(VALUE (GETMENUPROP MENU 'VALUE]
         (if VALUE
             then (SHADEITEM VALUE MENU (CONSTANT WHITESHADE]
    (PUTMENUPROP MENU 'VALUE ITEM)
    (SHADEITEM ITEM MENU (CONSTANT BLACKSHADE)])

(MMENU.BOLDITEM
  [LAMBDA (STRING FONT)                                          (* cdl "16-Oct-85 08:56")
    (LET [BITMAP STREAM (BOLDERFONT (FONTCOPY FONT 'WEIGHT 'BOLD]
         [SETQ STREAM (DSPCREATE (SETQ BITMAP (BITMAPCREATE (STRINGWIDTH STRING BOLDERFONT)
                                                     (FONTPROP FONT 'HEIGHT]
         (DSPFONT BOLDERFONT STREAM)
         (DSPYPOSITION (FONTPROP FONT 'DESCENT)
                STREAM)
         (PRIN1 STRING STREAM)
         BITMAP])

)

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

(ADDTOVAR NLAMA )

(ADDTOVAR NLAML )

(ADDTOVAR LAMA GRAPHCALLS)
)

(PUTPROPS GRAPHCALLS COPYRIGHT ("Stanford University" 1984 1985 1986 1987 1988))
```

FUNCTION INDEX

VARIABLE INDEX

PROPERTY INDEX

RECORD INDEX