File created: 1-Apr-2024 12:13:15 **{DSK}<home>frank>il>notecards>system>NCTEXTCARD.;51**

changes to: (FNS NC.TEditPutFn)

previous date: 31-Mar-2024 23:09:40 {DSK}<home>frank>il>notecards>system>NCTEXTCARD.;49

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

```
(RPAQQ NCTEXTCARDCOMS
       (

;;; Stuff for the Text card type.

        (FILES NCTYPESMECH TEDIT)
        (FNS NCAddStub.TextCard)
        (GLOBALVARS TEDIT.DEFAULT.MENU TEDIT.TITLED.ICON.TEMPLATE TEDIT.ICON.FONT NC.TEditPasswordDividedBy100
               NC.DefaultFont NC.DeletedLinkImageObject NC.CardTypes NC.AnnoFlg
               NC.UseDeletedLinkIconIndicatorsFlg NC.DelTEditProcessAtShrinkFlg NC.TextCardIcon
               NC.UserSpecifiedTEditProps NC.MODERNIZE.TITLEPROPORTION)
        (INITVARS (NC.MODERNIZE.TITLEPROPORTION 0.25))


;;; basic fns

        (FNS NC.MakeTEditCard NC.BringUpTEditCard NC.MakeTextCardReadOnly NC.MakeTextCardReadWrite
             NC.RevertTextCard NC.CollectReferencesInText NC.InstallTextTitleBarMiddleMenu NC.MarkTextDirty
             NC.ResetTEdit NC.ResetTEditProcess NC.TEditBasedP NC.TEditCloseFn NC.TEditMenuFn NC.TEditQuitFn
             NC.TextCopySubstance NC.TextDirtyP NC.TextCardShrinkFn NC.PutTextSubstance NC.GetTextSubstance
             NC.UpdateLinkImagesInText NC.CharPosFromTextStream NC.DelReferencesToCardFromText NC.TEditWindow
             NC.TEditCollectObjects)
                                                                      ; (DECLARE%: DONTEVAL@LOAD (VARS
                                                                      ; (TEDIT.DEFAULT.PROPS (CONS 'SLOWUPDATE (CONS 'T
                                                                      ; TEDIT.DEFAULT.PROPS)))))
        (INITVARS (NC.TEditPasswordDividedBy100 314))


;;; This stuff supports the push-copy method of copying links in NC

        (FNS NC.TranslateWindowPositionToTEditPosition NC.InsertLinkInTextWindow)


;;; New FNS

        (GLOBALVARS NC.ExternalPutLinkIconImageFns)
        (FNS NC.CardFromTextStream NC.TEditPutFn NC.TEditPutFnAuxiliary NC.TEditGetFn NC.MakeTEditPropsList
             NC.TEditPropsListQuitFn NC.DelTEditProcesses)


;;; Add the Text card type to the card types list.

        (FNS NC.AddTextCard NC.ComputeTextCardMiddleButtonMenuItems)
        (BITMAPS NC.TextCardIcon)
        [INITADVISE [TEDIT.ADD.MENUITEM (NIL (AFTER NIL (if (EQ MENU TEDIT.DEFAULT.MENU)
                                                            then
                                                            (NCP.ChangeCardTypeFields
                                                             'Text NIL `((MiddleButtonMenuItems ,(
                                                                        NC.ComputeTextCardMiddleButtonMenuItems
                                                                                                       ]
               (TEDIT.REMOVE.MENUITEM (NIL (AFTER NIL (if (EQ MENU TEDIT.DEFAULT.MENU)
                                                          then
                                                          (NCP.ChangeCardTypeFields
                                                           'Text NIL `((MiddleButtonMenuItems ,(
                                                                        NC.ComputeTextCardMiddleButtonMenuItems
                                                                                                     ]
        (DECLARE%: DONTEVAL@LOAD (P (NC.AddTextCard)))
        (PROP (FILETYPE MAKEFILE-ENVIRONMENT)
              NCTEXTCARD)))


;;; Stuff for the Text card type.

(FILESLOAD NCTYPESMECH TEDIT)

(DEFINEQ

(NCAddStub.TextCard
  [LAMBDA NIL                                                         (* kirk%: "19-Jun-86 20:52")

                (* * kirk 18Jun86 Add the text card stub)

    (NC.AddCardTypeStub 'Text 'NoteCard 'NCTEXTCARD NIL '((DisplayedInMenuFlg . T])
)
```

```
(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS TEDIT.DEFAULT.MENU TEDIT.TITLED.ICON.TEMPLATE TEDIT.ICON.FONT NC.TEditPasswordDividedBy100
       NC.DefaultFont NC.DeletedLinkImageObject NC.CardTypes NC.AnnoFlg NC.UseDeletedLinkIconIndicatorsFlg
       NC.DelTEditProcessAtShrinkFlg NC.TextCardIcon NC.UserSpecifiedTEditProps NC.MODERNIZE.TITLEPROPORTION)
)

(RPAQ? NC.MODERNIZE.TITLEPROPORTION 0.25)


;;; basic fns

(DEFINEQ

(NC.MakeTEditCard
  [LAMBDA (Card Title NoDisplayFlg ParamList InterestedWindow RegionOrPosition)
                                                                     ; Edited 11-Jun-88 15:58 by Trigg
                                                                     ; Make up a blank text card and display it.

    ;; rht 9/10/85: Fixed call to NC.MakeTEditPropsList so prompt window gets fixed up before card is brought up.

    ;; fgh 11/13/85 Updated to handle Card object.

    ;; rht 4/11/86: Now passes type to NC.MakeTEditMiddleMenu.

    ;; rht 5/6/86: Now calls NC.InstallTitleBarLeftMenu and NC.InstallTextTitleBarMiddleMenu.

    ;; rht 9/8/86: No longer hangs Card off NoteCardObject windowprop.  THis is done in NC.MakeNoteCard.

    ;; rht 9/19/86: Mod to above fix.  Now calls NC.MakeNewCardWindow a new fn that creates window and hangs Card off windowprop.  Also takes
    ;; ParamList arg.

    ;; rht&pmi 11/19/86: Now passes NoteFile arg to NC.MakeTEditPropsList.

    ;; rht 4/17/87: Now makes a prompt window before opening the window.

    ;; dsj 9/22/87: Added call to NC.SetTitle so title passing from calling fns works properly.

    ;; pmi 12/10/87: Added dsj's changes;  see above comments.

    ;; pmi 2/9/88: Missed a call to NC.SetTitle - it's in there now.

    ;; rht 6/9/88: Added RegionOrPosition arg and passed to NC.MakeNewCardWindow.  Also InterestedWindow arg.

    (LET (EmptyStream TextStream Window Type)
         (COND
            (NoDisplayFlg (SETQ EmptyStream (OPENSTREAM '{NODIRCORE} 'BOTH 'NEW))
                   [SETQ TextStream (OPENTEXTSTREAM EmptyStream NIL NIL NIL (NC.MakeTEditPropsList
                                                                             NIL
                                                                             (LISTGET ParamList
                                                                                    '
                                                                               Don'tAttachUserSpecifiedPropsFlg
                                                                                    ]
                   (NC.SetSubstance Card TextStream)
                   (NC.SetTitle Card Title)
                   (STREAMPROP TextStream 'NoteCardObject Card)
                   Card)
            (T (SETQ Window (NC.MakeNewCardWindow Card (OR Title "Untitled")
                                   NIL T RegionOrPosition))

               ;; Make a prompt window 'invisibly' so it gets hooked in.

               (NC.AttachPromptWindowOffScreen Window)
               (NC.SetTitle Card Title)
               (SETQ Type (NC.RetrieveType Card))
               (NC.InstallTitleBarLeftMenu Window Type)
               (NC.InstallTextTitleBarMiddleMenu Window Type)
               (WINDOWADDPROP Window 'SHRINKFN (FUNCTION NC.TextCardShrinkFn))
               (WINDOWPROP Window 'MODERNIZE.TITLEPROPORTION NC.MODERNIZE.TITLEPROPORTION)
               (TEDIT NIL Window NIL (NC.MakeTEditPropsList Window (NC.FileBoxP Card T)
                                            (fetch (Card NoteFile) of Card)))
               (until (WINDOWPROP Window 'TEXTSTREAM) do (BLOCK))
               (SETQ TextStream (WINDOWPROP Window 'TEXTSTREAM))
               (NC.SetSubstance Card TextStream)
               (STREAMPROP TextStream 'NoteCardObject Card)
               Window])

(NC.BringUpTEditCard
  [LAMBDA (Card TextStream RegionOrPosition)                         ; Edited 28-Apr-88 14:51 by Trigg

    ;; Bring up a TEdit window for Card ID whose text stream is TextStream in Region specified by RegionOrPosition or by the user.

    ;; rht 9/10/85: Fixed call to NC.MakeTEditPropsList so prompt window gets fixed up before card is brought up.

    ;; fgh 11/13/85 Updated to handle Card objects.

    ;; kirk 2Dec85 Removed illegal RETURN

    ;; rht 4/11/86: Added Type arg to call to NC.MakeTEditMiddleMenu and changed names of menu windowprops.

    ;; rht 5/5/86: Now calls NC.InstallTitleBarLeftMenu and NC.InstallTextTitleBarMiddleMenu.

    ;; rht 8/2/86: Now doesn't throw in any extra tedit props if card is a filebox.

    ;; rht 11/16/86: Now moves window to RegionOrPosition if already up.

    ;; rht&pmi 11/19/86: Now passes NoteFile arg to NC.MakeTEditPropsList.

    ;; rht & pmi 12/16/86: Changed the way we get TEditWindow.
    ;; rht 1/30/87: Now calls RESTART-PROCESS-OF-TEDIT-WINDOW from Bagley's TEDIT-PROCESS-KILLER package.

    ;; rht 4/17/87: Now makes a prompt window before opening the window.

    ;; rg 11/6/87 now waits for TEdit process to get going after TEDIT call

    ;; pmi 2/9/88: Now checks to see if RegionOrPosition is different from current region or position of card before calling SHAPEW, which could
    ;; needlessly repaint the window otherwise.

    ;;  pmi 2/18/88: Removed one of the two FLASHWs.  Fixed up check for same position to be a little more efficient.

    ;; rht 4/28/88: Removed call to NC.MakeTEditPropsList from within call to TEDIT.  That call already happens in NC.GetTextSubstance.

    (LET (Region TEditWindow WindowRegion TEditProcess Title TextStreamDirtyFlg Type TEditProcess)
         (if (SETQ TEditWindow (NC.FetchWindow Card))
             then (TOTOPW TEditWindow)
                  [if RegionOrPosition
                      then (if (REGIONP RegionOrPosition)
                               then (if [NOT (EQUAL RegionOrPosition (WINDOWPROP TEditWindow 'REGION]
                                        then (SHAPEW TEditWindow (NC.DetermineDisplayRegion Card RegionOrPosition)
                                                    ))
                             elseif (POSITIONP RegionOrPosition)
                               then (SETQ WindowRegion (WINDOWPROP TEditWindow 'REGION))
                                    (if [NOT (AND (EQUAL (fetch (POSITION XCOORD) of RegionOrPosition)
                                                         (fetch (REGION LEFT) of WindowRegion))
                                                  (EQUAL (fetch (POSITION YCOORD) of RegionOrPosition)
                                                         (fetch (REGION BOTTOM) of WindowRegion]
                                        then (SHAPEW TEditWindow (NC.DetermineDisplayRegion Card RegionOrPosition]
                  (FLASHW TEditWindow)
                  (if (SETQ TEditProcess (WINDOWPROP TEditWindow 'PROCESS))
                      then (TTY.PROCESS TEditProcess)
                    else                                              ; Process may have been turned off.  Try to restart.
                        (SETQ TEditProcess (RESTART-PROCESS-OF-TEDIT-WINDOW TEditWindow)))
                  TEditWindow
           else (SETQ Region (NC.DetermineDisplayRegion Card RegionOrPosition))
                (SETQ Title (NC.RetrieveTitle Card))
                (SETQ Type (NC.RetrieveType Card))
                (SETQ TEditWindow (CREATEW Region Title NIL T))
                (SETQ TextStreamDirtyFlg (NC.CardDirtyP Card))

                ;; Make a prompt window 'invisibly' so it gets hooked in.

                (NC.AttachPromptWindowOffScreen TEditWindow)
                (WINDOWADDPROP TEditWindow 'SHRINKFN (FUNCTION NC.TextCardShrinkFn))
                (WINDOWPROP TEditWindow 'TEDIT.TITLEMENUFN (FUNCTION NC.TEditMenuFn))
                (WINDOWPROP TEditWindow 'MODERNIZE.TITLEPROPORTION NC.MODERNIZE.TITLEPROPORTION)
                (NC.InstallTitleBarLeftMenu TEditWindow Type)
                (NC.InstallTextTitleBarMiddleMenu TEditWindow Type)
                (SETQ TEditProcess (TEDIT TextStream TEditWindow NIL (NC.MakeTEditPropsList TEditWindow)))

                ;; wait for process to get established.

                ;; Otherwise, the TEdit killer, called by MakeReadOnly, may fail

                (until (TEDIT-PROCESS-P TEditProcess) do (BLOCK))
                (if TextStreamDirtyFlg
                    then (NC.MarkCardDirty Card)))
         TEditWindow])

(NC.MakeTextCardReadOnly
  [LAMBDA (Card)                                                      (* Randy.Gobbel " 2-Dec-87 18:13")
    (DECLARE (GLOBALVARS NC.ShowPropListMenu))
    (LET ((Window (NC.FetchWindow Card))
          PropListEditor)
         (NC.ProtectedCardOperation Card "Make Read-Only" Window
                (if (NC.CardSomehowDirtyP Card)
                    then (ALLOW.BUTTON.EVENTS)
                         (if (MENU (create MENU
                                          TITLE _ "Card has been edited"
                                          ITEMS _ '(("Save Changes" T "Save edited version of card in NoteFile")
                                                    ("Revert" NIL "Discard edits, refetch card contents from
                                                           NoteFile"))
                                          CENTERFLG _ T))
                             then (NC.CardSaveFn Card NIL Window)
                           else (NC.RevertTextCard Card)))
                (NC.SetUserDataProp Card 'ReadOnly T)
                (if (WINDOWP (SETQ PropListEditor (NC.PropListEditorOpenP Window)))
                    then (NC.MakeTEditReadOnly PropListEditor)
                         (WINDOWPROP PropListEditor 'TEDIT.MENU NC.ShowPropListMenu))
                (if (WINDOWP (SETQ PropListEditor (NC.PropListEditorOpenP Window T)))
                    then (NC.MakeTEditReadOnly PropListEditor))
                (NC.MakeTEditReadOnly Window '(Assign% Title Title/FileBoxes |Add Global Link| Insert% Link
                                                      |Put Cards Here| Restart% Editor)
                       '(Delete% Card])

(NC.MakeTextCardReadWrite
  [LAMBDA (Card)                                                      (* Randy.Gobbel "23-Nov-87 14:11")
    (DECLARE (GLOBALVARS NC.EditPropListMenu))
    (LET ((Window (NC.FetchWindow Card))
          PropListEditor)
         (NC.ProtectedCardOperation Card "Make Read-Write" Window (if (WINDOWP (SETQ PropListEditor (
                                                                                       NC.PropListEditorOpenP
                                                                                                 Window)))
                                                                          then (NC.MakeTEditReadWrite PropListEditor
                                                                                    )
                                                                               (WINDOWPROP PropListEditor
                                                                                      'TEDIT.MENU
                                                                                      NC.EditPropListMenu))
                (if (WINDOWP (SETQ PropListEditor (NC.PropListEditorOpenP Window T)))
                    then (NC.MakeTEditReadWrite PropListEditor))
                (NC.MakeTEditReadWrite Window)
                (NC.SetUserDataProp Card 'ReadOnly NIL])

(NC.RevertTextCard
  [LAMBDA (Card)                                              (* Randy.Gobbel " 7-Dec-87 12:04")
    (NC.GetNoteCard Card)
    (LET (TEditProcess (Window (NC.FetchWindow Card)))
         (UNINTERRUPTABLY
             (KILL-PROCESS-OF-TEDIT-WINDOW Window)
             (UNMARK-AS-WITHOUT-PROCESS Window))
         [SETQ TEditProcess (TEDIT (NC.FetchSubstance Card)
                                   Window NIL (NC.MakeTEditPropsList Window (NC.FileBoxP Card T)
                                                     (fetch (Card NoteFile) of Card]
         (until (TEDIT-PROCESS-P TEditProcess) do (BLOCK)))   (* wait for process to establish itself so TEdit killer doesn't get
                                                                 confused)
    (NC.MarkCardDirty Card T])

(NC.CollectReferencesInText
  [LAMBDA (Card CheckAndDeleteFlg ReturnLinkIconsFlg ReturnLocationsFlg)
                                                              (* rht%: " 1-Jun-87 22:11")

          (* * Return a list of all links or link icons in text substance Substance.
          If CheckAndDeleteFlg, then delete any links found that are not valid links.)

          (* * fgh |11/13/85| Updated to handle Card object. Eliminated DatabaseStream argument.)

          (* * rht 5/26/87%: Changed to match reduced functionality of NC.ValidLinkP, now have to check that destination of
          ActualLink is a valid card.)

    (DECLARE (GLOBALVARS NC.DeletedLinkImageObject))
    (PROG (ActualLink DirtyFlg Links (Substance (NC.FetchSubstance Card)))
          [SETQ Links (for ImageObjectDecriptor in (NC.TEditCollectObjects Substance (FUNCTION NC.LinkIconImageObjP))
                         when (PROGN (SETQ ActualLink (NC.FetchLinkFromLinkIcon (CAR ImageObjectDecriptor)))
                                     (COND
                                        ((NULL CheckAndDeleteFlg)
                                                              (* No checking required)
                                         T)
                                        ((AND (LISTP CheckAndDeleteFlg)
                                              (FMEMB (fetch (Link DestinationCard) of ActualLink)
                                                     CheckAndDeleteFlg))
                                                              (* Already checked since ID cached on CheckAndDeleteFlg list)
                                         T)
                                        ((AND (NC.ValidLinkP ActualLink)
                                              (NC.ValidCardP (fetch (Link DestinationCard) of ActualLink)))
                                                              (* Link is valid)
                                         T)
                                        (T                    (* Link is bad. Replace it with the DeletedLink image object.)
                                           (create IMAGEOBJ smashing (CAR ImageObjectDecriptor)
                                                              OBJECTDATUM _ (fetch (IMAGEOBJ OBJECTDATUM)
                                                                                of NC.DeletedLinkImageObject)
                                                              IMAGEOBJPLIST _ (fetch (IMAGEOBJ IMAGEOBJPLIST)
                                                                                  of NC.DeletedLinkImageObject
                                                                                     )
                                                              IMAGEOBJFNS _ (fetch (IMAGEOBJ IMAGEOBJFNS)
                                                                                of NC.DeletedLinkImageObject))
                                           (SETQ DirtyFlg T)
                                           NIL)))
                         collect (COND
                                    ((AND ReturnLinkIconsFlg (NOT ReturnLocationsFlg))
                                     (CAR ImageObjectDecriptor))
                                    ((AND ReturnLinkIconsFlg ReturnLocationsFlg)
                                     (CONS (CAR ImageObjectDecriptor)
                                           (CADR ImageObjectDecriptor)))
                                    ((AND (NOT ReturnLinkIconsFlg)
                                          ReturnLocationsFlg)
                                     (CONS ActualLink (CADR ImageObjectDecriptor)))
                                    (T ActualLink]
          (RETURN (CONS Links DirtyFlg])

(NC.InstallTextTitleBarMiddleMenu
  [LAMBDA (Window CardType)                                   (* pmi%: " 2-Apr-87 10:23")

          (* * Make a text card middle button title bar menu and install.)
```

```
               (* * pmi 4/1/87%: Changed to replace MENUFONT with NC.MenuFont)

     (DECLARE (GLOBALVARS NC.MenuFont))
     (LET (Menu)
          [WINDOWPROP Window 'TitleBarMiddleButtonMenu (SETQ Menu (\TEDIT.CREATEMENU (NC.GetCardTypeField
                                                                                        MiddleButtonMenuItems
                                                                                        CardType]
          (replace (MENU MENUFONT) of Menu with NC.MenuFont)
          (replace (MENU ITEMHEIGHT) of Menu with (IPLUS (FONTPROP NC.MenuFont 'HEIGHT)
                                                         1))
          (replace (MENU IMAGE) of Menu with NIL])

(NC.MarkTextDirty
  [LAMBDA (Card ResetFlg)                                        (* fgh%: "13-Nov-85 23:19")

               (* * Mark or unmark TextStream as being changed.)

               (* * rht 2/1/85%: Updated to use the documented function.)

               (* * fgh |11/13/85| Updated to use Card)

     (TEDIT.STREAMCHANGEDP (TEXTSTREAM (NC.FetchSubstance Card))
            ResetFlg])

(NC.ResetTEdit
  [LAMBDA (TextStream)                                           (* NoteCards% User "17-Jun-84 01:58")

               (* * Kill the Tedit process running on TextStream and the restart a new one.
               Goal is to clean up display if TEdit goea awary.)

     (ADD.PROCESS (LIST (FUNCTION NC.ResetTEditProcess)
                        TextStream)
            'NAME
            'ResetTEdit])

(NC.ResetTEditProcess
  [LAMBDA (TextStream)                                           (* fgh%: " 7-Jun-86 19:16")

               (* * Added-process that actually does the works for NC.ResetTEdit.
               Kill the Tedit process running on TextStream and the restart a new one.
               Goal is to clean up display if TEdit goea awary.)

               (* * rht 2/1/85%: Changed to calls to NC.MarkCardDirty and NC.CardDirtyP.)

               (* * rht 10/8/85%: Now closes prompt window if TEDIT call brought it up.)

               (* * fgh |11/13/85| Updated to handle Card object.)

               (* * fgh |6/7/86| Upodated to use NC.GETPROMPTWINDOW. Added WINDOWADDPROP call to make sure that
               NC.QUitCard is first on the CLOSEFN list)

     (LET ((Window (NC.TEditWindow TextStream))
           DirtyFlg Card PromptWin)
          (COND
             ((AND (WINDOWP Window)
                   (SETQ Card (NC.CoerceToCard Window)))
              (SETQ DirtyFlg (NC.CardDirtyP Card))
              (TEDIT.KILL TextStream)
              (until (NULL (WINDOWPROP Window 'TEXTSTREAM)) do (BLOCK))
              (TEDIT TextStream Window NIL (NC.MakeTEditPropsList Window))
              (until (TEXTSTREAMP (WINDOWPROP Window 'TEXTSTREAM)) do (BLOCK))
              (COND
                 ((OPENWP (SETQ PromptWin (NC.GETPROMPTWINDOW Window NIL NIL T)))
                  (CLOSEW PromptWin)))
              (NC.MarkCardDirty Card (NOT DirtyFlg))
              (WINDOWPROP Window 'TITLE (NC.RetrieveTitle Card))
              (WINDOWADDPROP Window 'CLOSEFN (FUNCTION NC.QuitCard)
                     T])

(NC.TEditBasedP
  [LAMBDA (CardOrTypeName)                                       (* fgh%: "17-Nov-85 19:21")

               (* * Returns T if Type is a note card type that is based on TEdit or else an ID of such a note card.
               NIL otherise.)

               (* * rht 11/8/84%: Hacked to reflect notecards typing mechanism.
               Seems gross to have to search the types list to get the record.)

               (* * fgh |11/13/85| Updated to handle Card object.)

     (LET [(TypeName (COND
                        ((NC.CardP CardOrTypeName)
                         (NC.RetrieveType CardOrTypeName))
                        (T CardOrTypeName]
          (NC.IsSubTypeOfP TypeName 'Text])

(NC.TEditCloseFn
  [LAMBDA (CardIdentifier)                                ; Edited 30-Mar-88 17:45 by Trigg
                                                          ; Close this TEdit window by Quitting from TEdit

    ;; rht 11/12/84: Changed call to TEXTSTREAM to a WINDOWPROP call.

    ;; fgh 11/13/85 Updated to handle Card objects.

    ;; rht 8/8/87: Now smashes value of PREVPREVPC and its proplist so PIECE objects can be GC'ed.

    ;; rht 3/30/88: Now restarts tedit process if it was killed - this seems to prevent storage leak.

     (DECLARE (GLOBALVARS PREVPREVPC))
     (LET ((Card (NC.CoerceToCard CardIdentifier))
           Window)
          (SETQ Window (NC.FetchWindow Card))
          (NC.TEditQuitFn Card)
          [AND (WINDOWP Window)
               (PROGN (OR (WINDOWPROP Window 'PROCESS)             ; Process may have been turned off.  Try to restart.
                              (RESTART-PROCESS-OF-TEDIT-WINDOW Window))
                      (WINDOWDELPROP Window 'CLOSEFN (FUNCTION NC.QuitCard))
                      (TEDIT.QUIT (TEXTSTREAM Window]

          ;; Ugliness supremo: This to allow PIECE objects, and the links they point to, to be GC'ed.  Why does TEdit let this junk hang around???

          (SETQ PREVPREVPC NIL)
          (SETPROPLIST 'PREVPREVPC NIL])

(NC.TEditMenuFn
  [LAMBDA (Window)                                        (* Randy.Gobbel "23-Apr-87 15:32")

               (* * Gets called from LEFT or MIDDLE button press in title bar of TEdit window.
               If LEFT press bring up the Notecards Manipulation Window. IF MIDDLE press bring up the TEdit specific NoteCard
               operations menu. Bring up menus using TEDIT.DEFAULT.MENUFN)

               (* * rht 4/11/86%: No only calls TEDIT.DEFAULT.MENUFN for middle button press.
               On left button press, works like graph and sketch cards.)

               (* * rg |4/23/87| added contention lock)

     (if (LASTMOUSESTATE LEFT)
         then (APPLY* (OR (MENU (WINDOWPROP Window 'TitleBarLeftButtonMenu))
                          (FUNCTION NILL))
                      Window)
       else (WINDOWPROP Window 'TEDIT.MENU (WINDOWPROP Window 'TitleBarMiddleButtonMenu))
            (NC.ProtectedCardOperation (NC.CoerceToCard Window)
                  "Perform TEdit Operation" Window (TEDIT.DEFAULT.MENUFN Window])

(NC.TEditQuitFn
  [LAMBDA (CardIdentifier)                                (* rht%: "14-Jul-86 00:09")

               (* Called by TEdit when quitting out of a TEdit-based note card.
               Close up attached windows,)

               (* * fgh |11/13/85| Updated to handle Card objects.)

               (* * fgh |5/2/86| Removed assumption that there is a window when looking for the prompt window)

               (* * rht 7/14/86%: No longer calls NC.DeactivateCard.)

     (LET ((Card (NC.CoerceToCard CardIdentifier))
           PromptWindow Window)
          (SETQ Window (NC.FetchWindow Card))
          (SETQ PromptWindow (GETPROMPTWINDOW Window NIL NIL T))
          [AND Window (for AttachedWindow in (ALLATTACHEDWINDOWS Window) unless (EQ AttachedWindow PromptWindow)
                         do (DETACHWINDOW AttachedWindow)
                            (CLOSEW AttachedWindow)
                            (until (NULL (OPENWP AttachedWindow)) do (BLOCK]

               (* * Obsolete (* MAke sure TEdit won't close the database file) (replace
               (TEXTOBJ TXTFILE) of (TEXTOBJ TextStream) with NIL) (TEDIT.MAPPIECES
               (TEXTOBJ TextStream) (FUNCTION (LAMBDA (CH# PC PC# OBL)
               (replace (PIECE PFILE) of PC with NIL)))))

               (* * Setting the PromptWindow PROCESS to NIL is to break a circularity caused by TEXTOBJ -> PROMPTWINDOW ->
               PROCESS -> TEXTSTREAM -> TEXTOBJ)

          (AND Window PromptWindow (WINDOWPROP PromptWindow 'PROCESS NIL)
               (REMOVEPROMPTWINDOW Window))
          T])

(NC.TextCopySubstance
  [LAMBDA (Card FromStream ToStream Length)               (* fgh%: "21-Nov-85 21:26")
```

```
            (* * Copy a text substance from FromStream to ToStream.)

            (* * fgh |11/20/85| NoteCards now passes the Start and End pointers of the FromStream as args and properly sets the
            pointers on the ToStream)

            (* * fgh |11/21/85| Now passed Length instead of start and end ptrs.)

    (LET* ((FromStartPtr (GETFILEPTR FromStream))
           (FromEndPtr (PLUS Length FromStartPtr)))
          (COPYBYTES FromStream ToStream FromStartPtr FromEndPtr))
    T])
```

```
(NC.TextDirtyP
  [LAMBDA (Card)                                                      (* fgh%: "13-Nov-85 23:19")

            (* * Return T is TextSubstance has been changed.)

            (* * fgh |11/13/85| Updated to handle Card objects.)

    (TEDIT.STREAMCHANGEDP (TEXTSTREAM (NC.FetchSubstance Card])
```

```
(NC.TextCardShrinkFn
  [LAMBDA (W)                                                         ; Edited  2-Dec-88 15:48 by krivacic

;;; This is in order to get the right title on the TEdit icon.  This is a kludgy finesse around TEdit's shrunken icon title facility.

    ;; rht 4/30/86: Now optionally kills TEdit process at shrink time.

    ;; fgh 6/5/86 Fixed to KLUDGE around problem that TEdit will call GETBOXPOSITION even when SHRINKW is called with a position argument.
    ;; Kludge involves looking up the stack for the position being specified in SHRINKW since its not passed down to here.

    ;; rg 11/18/86 Added GLOBALVARS

    ;; rht 1/30/87: Changed NC.DelTEditProcessAtShrinkFlg stuff to use Bagley's package via KILL-PROCESS-OF-TEDIT-WINDOW.

    ;; rht 4/24/88: Now trashes TEDIT.ICON.TITLE prop to be NIL rather than "" to prevent \TEDIT.SHRINK.ICONCREATE from undoing our work.

    (DECLARE (GLOBALVARS TEDIT.ICON.FONT TEDIT.TITLED.ICON.TEMPLATE NC.DelTEditProcessAtShrinkFlg NC.IconProps)
           )                                                          (* LET ((OldIconTitle (WINDOWPROP W
                                                                      (QUOTE NoteCardsIconTitle))) (Icon
                                                                      (WINDOWPROP W (QUOTE ICON)))
                                                                      (Card (NC.CoerceToCard W)) IconTitle)
                                                                      (if (NULL Icon) then (WINDOWPROP W
                                                                      (QUOTE TEDIT.ICON.TITLE) NIL)
                                                                      (WINDOWPROP W (QUOTE ICON)
                                                                      (OR (LET ((CARD-ICON-INFO (LISTGET NC.IconProps
                                                                      (fetch (CardObject Type) of (WINDOWPROP W
                                                                      (QUOTE NoteCardObject)))))) (AND CARD-ICON-INFO
                                                                      (TITLEDICONW CARD-ICON-INFO "" NC.IconFont
                                                                      (LET ((StkPos (STKPOS (QUOTE SHRINKW))))
                                                                      (if StkPos then (PROG1 (STKARG
                                                                      (QUOTE ICONPOSITION) StkPos)
                                                                      (RELSTK StkPos)) else NIL)) T T NIL)))
                                                                      (TITLEDICONW TEDIT.TITLED.ICON.TEMPLATE ""
                                                                      TEDIT.ICON.FONT (LET ((StkPos
                                                                      (STKPOS (QUOTE SHRINKW))))
                                                                      (if StkPos then (PROG1 (STKARG
                                                                      (QUOTE ICONPOSITION) StkPos)
                                                                      (RELSTK StkPos)) else NIL)) T T NIL)))
                                                                      (APPLY* (WINDOWPROP W (QUOTE ICONFN)) W)
                                                                      (SETQ Icon (WINDOWPROP W (QUOTE ICON))))
                                                                      (if (NOT (EQUAL OldIconTitle (SETQ IconTitle
                                                                      (CONCAT "NC: " (NC.RetrieveTitle Card))))) then
                                                                      (ICONTITLE IconTitle NIL NIL Icon)
                                                                      (WINDOWPROP W (QUOTE NoteCardsIconTitle) IconTitle))
                                                                      (* ;; " Kill tedit process if desired.")
                                                                      (if NC.DelTEditProcessAtShrinkFlg then
                                                                      (KILL-PROCESS-OF-TEDIT-WINDOW W)))

    ;; Generic Window Shrink

    (NC.ShrinkIcon W)

    ;;  Kill tedit process if desired.

    (if NC.DelTEditProcessAtShrinkFlg
        then (KILL-PROCESS-OF-TEDIT-WINDOW W])
```

```
(NC.PutTextSubstance
  [LAMBDA (Card Stream)                                               (* fgh |1/22/2024| Updated for 2023-24 TEdit revamp)
                                                                      (* rht%: "20-Aug-86 18:46")
                                                                      (* Put text substance from card ID on the database file)

            (* * rht 9/13/85%: Now doesn't try to write piece table at all if substance has zero length.)

            (* * fgh |11/13/85| Updated to handle NoteFile, Card objects.)

            (* * fgh |11/20/85| removed code to handle start and end pointers)
```

```
          (* * rht 1/23/86%: Now takes Stream as arg instead of computing from Card.)

          (* * fgh |2/6/86| Now returns a version number.)

          (* * Now checks to make sure that TEXTLEN > 0 before FORMATTEDP call.)

          (* * rht 8/20/86%: No longer trust TEDIT.FORMATTEDFILEP. Now decide whether there’s formatting info based on how
          many bytes were written down by TEDIT.PUT.PCTB.)

    (LET [(Substance (NC.FetchSubstance Card))
          (TmpStrm (OPENSTREAM ’{NODIRCORE} ’BOTH]
         (if (EQ (TEDIT.NCHARS Substance)
                 0)
             then (TEDIT.INSERT Substance " " 1))
         (TEDIT.PUT.STREAM Substance TmpStrm NIL NIL T)
         (SETFILEPTR TmpStrm 0)
         (COPYBYTES TmpStrm Stream)
         (SETFILEPTR Stream -1)
         0])

(NC.GetTextSubstance
  [LAMBDA (Card Length Stream)                                       (* rht%: " 2-Feb-87 17:51")
                                                                     (* Get a text stream from the database file)

          (* * fgh |11/13/85| Updated to handle Card objects.)

          (* * fgh |11/20/85| NoteCards now passes down start and end pointer as args.)

          (* * fgh |11/21/85| Now passed Length instead of start and end ptrs.)

          (* * rht 1/23/86%: Now takes Stream as arg instead of computing from Card.)

          (* * rht 2/2/87%: Fixed bug in the way NC.MakeTEditPropsList was being called.)

    (LET ((TempStream (OPENSTREAM ’{NODIRCORE} ’BOTH ’NEW))
          TextStream StartPtr)

          (* * Copy text stream to a NODIRCORE file from the current location.
          Number of bytes copied should be Length minus the 4 bytes already read for OriginalLocation)

         (COPYBYTES Stream TempStream (SETQ StartPtr (GETFILEPTR Stream))
                (PLUS StartPtr Length))

          (* * return an open textstream on the temp file)

         [SETQ TextStream (OPENTEXTSTREAM TempStream NIL NIL NIL (NC.MakeTEditPropsList NIL (NC.FileBoxP Card T)
                                                                        (fetch (Card NoteFile) of Card]
         (STREAMPROP TextStream ’NoteCardObject Card)
         TextStream])

(NC.UpdateLinkImagesInText
  [LAMBDA (SourceCardOrWindow DestinationCard)                       (* rht%: "28-Oct-87 18:47")

          (* * Update the notecard link image objects that point to DestinationID fromn text card SourceCardIDOrWindow)

          (* * rht 2/8/85%: Makes sure not to dirty the card if it wasn’t dirty before.)

          (* * fgh |11/17/85| Updated to handle card objects.)

          (* * rht 10/28/87%: Updated to handle pointer imageobjs as well.)

    (LET (Window TextObject LinksAndPointers SourceCard WasDirtyFlg)
         [SETQ Window (COND
                         ((WINDOWP SourceCardOrWindow)
                          SourceCardOrWindow)
                         (T (NC.FetchWindow SourceCardOrWindow]
         (SETQ TextObject (WINDOWPROP Window ’TEXTOBJ))
         [SETQ LinksAndPointers (NC.TEditCollectObjects TextObject (FUNCTION (LAMBDA (ImageObj)
                                                                       (OR (NC.LinkIconImageObjP ImageObj)
                                                                           (NC.PointerIconImageObjP
                                                                                  ImageObj]
         [SETQ WasDirtyFlg (NC.CardDirtyP (SETQ SourceCard (NC.CoerceToCard Window]
         [for LinkOrPointer in LinksAndPointers
            do (if (OR (AND (NC.LinkIconImageObjP (CAR LinkOrPointer))
                            (NC.SameCardP (fetch (Link DestinationCard) of (NC.FetchLinkFromLinkIcon (CAR
                                                                                                LinkOrPointer
                                                                                                    )))
                                   DestinationCard))
                       (AND (NC.PointerIconImageObjP (CAR LinkOrPointer))
                            (NC.SameCardP (fetch (NCPointer DestinationCard) of (NC.FetchPointerFromPointerIcon
                                                                                      (CAR LinkOrPointer)))
                                   DestinationCard)))
                   then (TEDIT.OBJECT.CHANGED TextObject (CAR LinkOrPointer]
         (NC.MarkCardDirty SourceCard (NOT WasDirtyFlg])
(NC.CharPosFromTextStream
  [LAMBDA (TEXTSTREAM)                                           (* rht%: "27-Jun-85 15:38")

              (* * Return the position %# of current selection in TextStream)

    (TEDIT.GETPOINT TEXTSTREAM])

(NC.DelReferencesToCardFromText
  [LAMBDA (SourceCard LinkOrDestinationCard Don'tCreateDeletedImageObjFlg)
                                                                 (* rht%: " 4-Nov-86 21:01")

              (* Delete all IMAGEOBJECTS in textstream specified by Substance that are Link Icons whose DESTINATIONID is eq to
              Destination ID)

              (* * rht 9/12/85%: Fixed so restores old selection after putting in deleted link icon.)

              (* * kirk%: 14Nov85%: changed EQs to NC.Sames. Deleted use of LinkID.)

              (* * rht 4/30/86%: Now only inserts NC.DeletedLinkImageObject if NC.UseDeletedLinkIconIndicators is non-nil.)

              (* * rht 11/4/86%: Now takes Don'tCreateDeletedImageObjFlg arg.)

    (DECLARE (GLOBALVARS NC.NoDeleteImageFns NC.UseDeletedLinkIconIndicatorsFlg NC.DeletedLinkImageObject))
    (LET ((LinkFlg (type? Link LinkOrDestinationCard))
          DestinationCard Substance)
         (COND
            [LinkFlg (OR (NC.CardP SourceCard)
                         (SETQ SourceCard (fetch (Link SourceCard) of LinkOrDestinationCard]
            (T (SETQ DestinationCard LinkOrDestinationCard)))
         (SETQ Substance (NC.FetchSubstance SourceCard))
         (for Link in (NC.TEditCollectObjects Substance (FUNCTION NC.LinkIconImageObjP)) bind OldSel
            when [COND
                    [LinkFlg (NC.SameLinkP LinkOrDestinationCard (NC.FetchLinkFromLinkIcon (CAR Link]
                    (T (NC.SameCardP DestinationCard (fetch (Link DestinationCard) of (NC.FetchLinkFromLinkIcon
                                                                                        (CAR Link]
            do (replace (IMAGEOBJ IMAGEOBJFNS) of (CAR Link) with NC.NoDeleteImageFns)
                                                                 (* WHENDELETEDFN taken care of earlier in delete cycle)
               (SETQ OldSel (TEDIT.GETSEL Substance))
               (TEDIT.DELETE Substance (CADR Link)
                      1)
               (AND NC.UseDeletedLinkIconIndicatorsFlg (NOT Don'tCreateDeletedImageObjFlg)
                    (TEDIT.INSERT.OBJECT NC.DeletedLinkImageObject Substance (CADR Link)))
               (TEDIT.SETSEL Substance OldSel])

(NC.TEditWindow
  [LAMBDA (TEditThing NoError)

    ;; Return the Window associated with a TEXTOBJ or TEXTSTREAM

    (LET ((TEditWindow (TEDITWINDOWP TEditThing)))
         (if (NULL TEditThing)
             then                                                (* Create the window, if none is given.)
                  (CREATEW NIL "Editing Window")
           elseif (AND (NULL TEditWindow)
                       (NULL NoError))
             then (ERROR TEditThing "is not associated with a TEdit window.")
           else TEditWindow])

(NC.TEditCollectObjects
  [LAMBDA (TEXTSTREAM TESTFN)                                    (* rrb " 8-Jun-84 11:12")
                                                                 (* fgh " 3-Mar-24" "Modified to use public Tedit API - no need to
                                                                 know about pieces etc")

              (* Map thru all the pieces in a text stream, and select the image objects paired with their character positions)

    (for FoundObj in (TEDIT.MAP.OBJECTS TEXTSTREAM [FUNCTION (LAMBDA (CH# OBJ FNARG)
                                                               (OR (NULL TESTFN)
                                                                   (APPLY* TESTFN OBJ]
                            NIL T)
       collect (LIST (CADR FoundObj)
                     (CAR FoundObj])
)

;; (DECLARE%: DONTEVAL@LOAD (VARS (TEDIT.DEFAULT.PROPS (CONS 'SLOWUPDATE (CONS 'T TEDIT.DEFAULT.PROPS)))))

(RPAQ? NC.TEditPasswordDividedBy100 314)

;;; This stuff supports the push-copy method of copying links in NC

(DEFINEQ

(NC.TranslateWindowPositionToTEditPosition
  [LAMBDA (Card Window WindowPositionX WindowPositionY)          (* fgh%: " 6-Feb-86 22:12")

            (* * fgh |2/6/86| Added Card argument.)

    (PROG (TextObj)
          (COND
             ([AND (WINDOWP Window)
                   (type? TEXTOBJ (SETQ TextObj (WINDOWPROP Window 'TEXTOBJ]
              (RETURN (fetch (SELECTION CH#) of (\TEDIT.SELECT WindowPositionX WindowPositionY TextObj
                                                         (fetch (TEXTOBJ MOUSEREGION) of TextObj)
                                                         NIL NIL Window])

(NC.InsertLinkInTextWindow
  [LAMBDA (Card TextWindow LinkOrLinkLabel CharacterPosition)    (* fgh%: " 6-Feb-86 22:12")

            (* * Given a Window and a Link, insert the Link at CharacterPosition in the TextStream of Window)

            (* * fgh |2/6/86| Added Card argument)

    (PROG [(TextStream (WINDOWPROP TextWindow 'TEXTSTREAM]
          (RETURN (NC.InsertLinkInText TextStream LinkOrLinkLabel NIL NIL NIL CharacterPosition])
)


;;; New FNS

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS NC.ExternalPutLinkIconImageFns)
)

(DEFINEQ

(NC.CardFromTextStream
  [LAMBDA (TextStream)                                           (* fgh%: "13-Nov-85 23:34")

            (* * Return the NoteCards ID for a TextStream)

            (* * fgh |11/13/85| Updated to handle Card object.)

    (STREAMPROP TextStream 'NoteCardObject])

(NC.TEditPutFn
  [LAMBDA (TextStream FileName When?)                            ; Edited  6-Jan-89 18:33 by rtk

;;;

;;; This function designed to be attached to the PUTFN Tedit property of NC textstreams.

;;; Before actions: Mark the card dirty for NoteCards, if TEdit thinks the card is dirty.  Also, change the link icons to be ExternalPut Link Icons so that
;;; thety are no longer active in the file being put to.  Necessary when a put is done outside the normal NoteCards mechanism.

;;; fgh 5/7/86 Added kludge to get around bug in TEDIT.PUT that causes FileName to be NIL most of the time.  Submitted AR #5621 about TEDIT.PUT
;;; bug.  Also added scheme to temporarily set the TEXPROP CACHE in order to prevent TEDIT.PUT from updating the file info in the piece table.

;;; BK 1/6/89 Add extra NC.ExternalBitMapPutLinkIconImageFns to NC.TEditPutFnAuxiliary call.

;;;

;;;  FGH 2024-03-31 Major rework of this function to accomodate changes in TEDIT.PUT in TEdit "4th round"

;;;                         Note that this function depends on TEDIT.PUT wrapping the call to this function inside a RESETLST.

;;;

    (if (EQ When? 'BEFORE)
        then ;;  In some cases, TEDIT passes in a TEXTOBJ instead of a TEXTSTREAM; convert it

             (SETQ TextStream (TEXTSTREAM TextStream))

             ;; first Mark the card dirty NoteCards style, if necessary

             [COND
                ((TEDIT.STREAMCHANGEDP TextStream)
                 (NC.MarkCardDirty (NC.CardFromTextStream (TEXTSTREAM TextStream]

             ;; Collect all the link icons in the TextStream and temporarily set them to be ExternalPut link icons

             [LET [(LinkIcons (NC.TEditCollectObjects TextStream (FUNCTION NC.LinkIconImageObjP]
                  (RESETSAVE (NC.TEditPutFnAuxiliary LinkIcons NC.ExternalPutLinkIconImageFns
                                    NC.ExternalBitMapPutLinkIconImageFns)
                         `(NC.TEditPutFnAuxiliary ,LinkIcons ,NC.LinkIconImageFns]

             ;; Fix up the title on the way out

             [RESETSAVE NIL `(AND ,(NC.TEditWindow TextStream)
                                  (WINDOWPROP ,(NC.TEditWindow TextStream)
                                         'TITLE
                                         ,(NC.RetrieveTitle (NC.CardFromTextStream TextStream]
                                                                  ;
             T
      else NIL])

(NC.TEditPutFnAuxiliary
  [LAMBDA (LinkIcons NewImageFns BitMapGetImageFns)                ; Edited  6-Jan-89 16:19 by rtk

;;; BK 1/6/89 Test for A Bitmap & Set to Alternage BitMapGetImageFns & TextStream

;;; For a list of link icons as returned by TEDIT.LIST.OF.OBJECTS, replace all the IMAGEFNS of the objects with NewImageFns

    (for LinkIconPair in LinkIcons
       do (LET ((IMAGE-OBJECT (CAR LinkIconPair)))
                (replace (IMAGEOBJ IMAGEOBJFNS) of IMAGE-OBJECT
                  with (if [AND BitMapGetImageFns (OR T (NC.SetBitMap&Box IMAGE-OBJECT (IMAGEOBJPROP IMAGE-OBJECT
                                                                                               'IMAGEBITMAP]
                           then BitMapGetImageFns
                         else NewImageFns])

(NC.TEditGetFn
  [LAMBDA (TextStream FileName When?)                              (* fgh%: "19-May-86 13:09")

          (* * This function called from the GETFN for all NC TextStreams)

          (* * fgh |5/19/86| Added WINDOWP check to make sure that the window title isn't changed if there is no window.)

    (SELECTQ When?
         (BEFORE (LET [(LinkIcons (NC.TEditCollectObjects TextStream (FUNCTION NC.LinkIconImageObjP]

          (* * Delete all the links currentluy in the text stream)

                      (for LinkIconPair in LinkIcons do (TEDIT.DELETE TextStream (CADR LinkIconPair)
                                                                       1))))
         (AFTER [LET ((Card (NC.CardFromTextStream TextStream)))

          (* * Mark the card dirty ala NoteCards)

                     (NC.MarkCardDirty Card)

          (* * Reset the NC title)

                     (AND (WINDOWP (NC.TEditWindow TextStream T))
                          (WINDOWPROP (NC.TEditWindow TextStream)
                                 'TITLE
                                 (NC.RetrieveTitle Card])
         NIL])

(NC.MakeTEditPropsList
  [LAMBDA (TEditWindow Don'tAttachUserSpecifiedPropsFlg NoteFile)
                                                                  (* rht%: " 9-Mar-87 11:00")

          (* * Create the props list to hand to all TEdit and OPENTEXTSTREAM calls)

          (* * rht 9/10/85%: Now takes a TEditWindow arg so can compute a prompt window.)

          (* * 5/1/86%: Now appends NC-specific props in front of TEdit props specified as global param by user.)

          (* * rht 8/2/86%: Added Don'tAttachUserSpecifiedPropsFlg to override attaching user-specified props.
          For use by fileboxes.)

          (* * rht&pmi 11/19/86%: Added another set of props at the notefile level stashed on NoteFile user data.
          Also added NoteFile argument.)

          (* * rht 3/9/87%: Now passes DON'T to PROMPTWINDOW to fix things for lyric.)

    (DECLARE (GLOBALVARS NC.UserSpecifiedTEditProps NC.DefaultFont))
    (APPEND (LIST 'TITLEMENUFN (FUNCTION NC.TEditMenuFn)
                  'PUTFN
                  (FUNCTION NC.TEditPutFn)
                  'GETFN
                  (FUNCTION NC.TEditGetFn)
                  'QUITFN
                  [LIST (FUNCTION NC.TEditPropsListQuitFn)
                         (FUNCTION (LAMBDA NIL
                                     'DON'T]
                  'PROMPTWINDOW
                  'DON'T)
           (LIST 'FONT (OR (AND (type? NoteFile NoteFile)
                                 (NC.NoteFileProp NoteFile 'Font))
                            NC.DefaultFont))
           (if Don'tAttachUserSpecifiedPropsFlg
               then NIL
             else (OR (AND (type? NoteFile NoteFile)
                           (NC.NoteFileProp NoteFile 'ExtraTEditProps))
                      NC.UserSpecifiedTEditProps))
           (LIST 'NOTSPLITTABLE T])


(NC.TEditPropsListQuitFn
  [LAMBDA (Window TextStream TextObj)                               (* fgh%: " 4-Jun-86 20:15")

          (* * QuitFn to be part of the props list when calling TEdit.)

          (* * fgh |6/4/86| First created.)

    (COND
       ((NC.CoerceToCard Window)
        (NC.QuitCard Window T))
       (T T])


(NC.DelTEditProcesses
  [LAMBDA (Window)                                                  (* rht%: "30-Apr-86 20:13")

          (* * Kill any TEdit processes running on Window and its attached windows.
          Notice recursive call to handle attached windows of attached windows.)

    (LET (Process)
         (if [PROCESSP (SETQ Process (WINDOWPROP Window 'PROCESS]
             then (DEL.PROCESS Process))
         (for AttachedWin in (ATTACHEDWINDOWS Window) do (NC.DelTEditProcesses AttachedWin])

)


;;; Add the Text card type to the card types list.

(DEFINEQ

(NC.AddTextCard
  [LAMBDA NIL                                                       ; Edited  3-Dec-87 19:01 by rht:

          (* * fgh |2/17/86| Added attached bit map field.)

          (* * rht 4/7/86%: Now you get all TEdit menu items on middle button as well as Restart% Editor.)

          (* * rg |11/20/87| added MakeReadOnlyFn & MakeReadWriteFn)

    (NC.AddCardType 'Text 'NoteCard `[(MakeFn ,(FUNCTION NC.MakeTEditCard))
                                       (EditFn ,(FUNCTION NC.BringUpTEditCard))
                                       (QuitFn ,(FUNCTION NC.TEditCloseFn))
                                       (MakeReadOnlyFn ,(FUNCTION NC.MakeTextCardReadOnly))
                                       (MakeReadWriteFn ,(FUNCTION NC.MakeTextCardReadWrite))
                                       (GetFn ,(FUNCTION NC.GetTextSubstance))
                                       (PutFn ,(FUNCTION NC.PutTextSubstance))
                                       (CopyFn ,(FUNCTION NC.TextCopySubstance))
                                       (MarkDirtyFn ,(FUNCTION NC.MarkTextDirty))
                                       (DirtyPFn ,(FUNCTION NC.TextDirtyP))
                                       (CollectLinksFn ,(FUNCTION NC.CollectReferencesInText))
                                       (DeleteLinksFn ,(FUNCTION NC.DelReferencesToCardFromText))
                                       (UpdateLinkIconsFn ,(FUNCTION NC.UpdateLinkImagesInText))
                                       (InsertLinkFn ,(FUNCTION NC.InsertLinkInTextWindow))
                                       (TranslateWindowPositionFn ,(FUNCTION
                                                                    NC.TranslateWindowPositionToTEditPosition]
           `((LinkDisplayMode Title)
             (DisplayedInMenuFlg T)
             (DefaultWidth 300)
             (DefaultHeight 200)
             (LinkAnchorModesSupported T)
             (LinkIconAttachedBitMap ,NC.TextCardIcon)
             (MiddleButtonMenuItems ,(NC.ComputeTextCardMiddleButtonMenuItems])


(NC.ComputeTextCardMiddleButtonMenuItems
  [LAMBDA NIL                                                       ; Edited 15-Dec-88 11:04 by krivacic

          (* * Using TEDIT.DEFAULT.MENU, build the list of middle button menu items for Text cards.)

    (DECLARE (GLOBALVARS TEDIT.DEFAULT.MENU))
    (APPEND (for Item in (fetch (MENU ITEMS) of TEDIT.DEFAULT.MENU)
               collect (if (EQ Item 'Quit)
                           then '(Quit (FUNCTION NC.QuitCard)
                                       "Close this note card after saving it in the NoteFile.")
                         else Item))
           '(BAR (Restart% Editor (FUNCTION NC.ResetTEdit)
                       "Resets and restarts the editor for this Card/Box."])

)
```

```
(RPAQQ NC.TextCardIcon )

[LOADINITADVISE NC-ADVICE0186 [TEDIT.ADD.MENUITEM
                               (NIL (AFTER NIL (if (EQ MENU TEDIT.DEFAULT.MENU)
                                                   then (NCP.ChangeCardTypeFields 'Text NIL
                                                               `((MiddleButtonMenuItems ,(
                                                                               NC.ComputeTextCardMiddleButtonMenuItems
                                                                                         ]
       (TEDIT.REMOVE.MENUITEM (NIL (AFTER NIL (if (EQ MENU TEDIT.DEFAULT.MENU)
                                                  then (NCP.ChangeCardTypeFields 'Text NIL
                                                              `((MiddleButtonMenuItems ,(
                                                                               NC.ComputeTextCardMiddleButtonMenuItems
                                                                                        ]

(DECLARE%: DONTEVAL@LOAD

(NC.AddTextCard)
)

(PUTPROPS NCTEXTCARD FILETYPE :FAKE-COMPILE-FILE)

(PUTPROPS NCTEXTCARD MAKEFILE-ENVIRONMENT (:PACKAGE "IL" :READTABLE "INTERLISP" :BASE 10))
```

FUNCTION INDEX

VARIABLE INDEX

PROPERTY INDEX