

File created: 7-Mar-2024 22:47:10 {DSK}<home>frank<il>notecards>library>NCTABLETOPCARD.;2

changes to: (FNS NCTableTop.ComputeCardsFromTemplate)

previous date: 5-Nov-2020 20:03:07 {DSK}<home>frank<il>notecards>library>NCTABLETOPCARD.;1

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

(RPAQQ NCTABLETOPCARDOMS

```
[(DECLARE%: DONTCOPY (PROPS (NCTABLETOPCARD MAKEFILE-ENVIRONMENT)
                             (NCTABLETOPCARD FILETYPE)))]
```

;;; Table top cards have substances containing card UIDs and screen regions. Following a link to a tabletop brings up its card in the stored regions.

```
(GLOBALVARS NCTableTop.LinkIconBitmap NCTableTop.OpenEventsExpr NCTableTop.OldOpenEventsExprs
            NCTableTop.MiddleButtonMenuItems)
[INITVARS [NCTableTop.OpenEventsExpr '(PROGN (NC.LoadFileFromDirectories 'NCTABLETOPCARD)
                                             (NCTableTop.NoteFileWhenOpenedFn NoteFile)
                                             [NCTableTop.OldOpenEventsExprs '( (PROGN (NC.LoadFileFromDirectories NCTABLETOP)
                                                                    (NCTableTop.NoteFileWhenOpenedFn NoteFile))
                                                                    (PROGN (FILESLOAD (FROM NOTECARDS)
                                                                    NCTABLETOP)
                                                                    (NCTableTop.NoteFileWhenOpenedFn NoteFile)
                                                                    (NCTableTop.MiddleButtonMenuItems '( (Recompute% TableTop (FUNCTION NCTableTop.RecomputeTableTop)
                                                                    "Recomputes this table top using current configuration
                                                                    of its cards."
                                                                    (SUBITEMS (New% Cards (FUNCTION
                                                                    NCTableTop.RemakeTableTopSelectingNewCards
                                                                    )
                                                                    "Recompute this table top using a new
                                                                    list of cards."
                                                                    (|New Cards In Order| (FUNCTION
                                                                    NCTableTop.RemakeTableTopSelectingNewCardsInOrder
                                                                    )
                                                                    "Recompute this table top using a new
                                                                    list of cards, in the order in which
                                                                    they are selected."
                                                                    (With% Template (FUNCTION
                                                                    NCTableTop.ComputeTableTopUsingTemplate
                                                                    )
                                                                    "Recompute this table top using a
                                                                    template table top."))
                                                                    (|Add cards to TableTop| (FUNCTION
                                                                    NCTableTop.AddSelectedCardsToTableTop
                                                                    )
                                                                    "Add cards to the table top.")
                                                                    (|Remove cards from TableTop| (FUNCTION
                                                                    NCTableTop.RemoveSelectedCards
                                                                    )
                                                                    "Remove cards from the table top.")
                                                                    (|Show cards in TableTop| (FUNCTION
                                                                    NCTableTop.OpenCardInTableTop
                                                                    )
                                                                    "Bring up one of the cards in the table top choosing
                                                                    from a menu of card titles."
                                                                    (|Bring up TableTop| (FUNCTION NCTableTop.BringUpTableTop)
                                                                    "Bring up the table top cards in proper layout.")
                                                                    (|Close down TableTop| (FUNCTION NCTableTop.CloseDownTableTop
                                                                    )
                                                                    "Close all cards in the table top.")]
```

;;; The following functions are exported, i.e. documented and callable from outside.

```
(FNS NCTableTop.BringUpTableTop NCTableTop.OpenCardInTableTop NCTableTop.CloseDownTableTop
     NCTableTop.RemakeTableTop NCTableTop.RecomputeTableTop NCTableTop.RemoveCardsFromTableTop
     NCTableTop.AddCardsToTableTop)
(FNS NCTableTop.CardsInTableTop NCTableTop.CollectCards)
(FNS NCTableTop.TableTopBasedP)
```

;;; The rest are internal.

```
(FNS NCTableTop.ComputeTableTop NCTableTop.ComputeTTEnterProps NCTableTop.ComputeCardsFromTemplate
     NCTableTop.BringUpCardInTableTop NCTableTop.ProcessTTEnterProps)
(FNS NCTableTop.RemakeTableTopSelectingNewCards NCTableTop.RemakeTableTopSelectingNewCardsInOrder
     NCTableTop.ComputeTableTopUsingTemplate NCTableTop.RemoveSelectedCards
     NCTableTop.AddSelectedCardsToTableTop)
(FNS NCTableTop.AddOpenEventsExpr NCTableTop.NoteFileWhenOpenedFn
     NCTableTop.MakeOrUpdateNoteFileTableTop NCTableTop.FixNoteFileOperationsMenus
     NCTableTop.CloseNoteFileSavingTableTop NCTableTop.CloseSessionSavingTableTops)
(FNS NCTableTop.MakeFn NCTableTop.EditFn NCTableTop.PutFn NCTableTop.GetFn)
(FNS NCTableTop.ReplaceUIDsFn)
```

```
(FNS NCTableTop.AddTableTopCard)
(BITMAPS NCTableTop.LinkIconBitmap)
(RECORDS TableTopEntry NCTableTop.BrowserOverviewSpecs)
(DECLARE%: DONTVAL@LOAD (P (NCTableTop.AddTableTopCard)
                           (PUTPROP 'TableTop 'ReplaceUIDsFn (FUNCTION NCTableTop.ReplaceUIDsFn))
                           (PUTPROP 'TableTop 'ExtraLinkIconMenuItems NCTableTop.MiddleButtonMenuItems)
                           (PUTPROP 'TableTop 'LinkIconMiddleButtonMenu NIL)))
(FNS NCAddStub.TableTopCard)
(DECLARE%: DONTVAL@LOAD (P (NCTableTop.FixNoteFileOperationsMenus]
                           ; Add a CloseSavingTableTop item to the Close notefile menus.
```

(DECLARE%: DONTCOPY

(PUTPROPS NCTABLETOPCARD MAKEFILE-ENVIRONMENT (:PACKAGE "INTERLISP" :READTABLE "INTERLISP"))

(PUTPROPS NCTABLETOPCARD FILETYPE :FAKE-COMPILE-FILE)
)

;;; Table top cards have substances containing card UIDs and screen regions. Following a link to a tabletop brings up its card in the stored regions.

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS NCTableTop.LinkIconBitmap NCTableTop.OpenEventsExpr NCTableTop.OldOpenEventsExprs
NCTableTop.MiddleButtonMenuItems)
)

(RPAQ? NCTableTop.OpenEventsExpr ' (PROGN (NC.LoadFileFromDirectories 'NCTABLETOPCARD)
(NCTableTop.NoteFileWhenOpenedFn NoteFile))

(RPAQ? NCTableTop.OldOpenEventsExprs ' ((PROGN (NC.LoadFileFromDirectories NCTABLETOP)
(NCTableTop.NoteFileWhenOpenedFn NoteFile))
(PROGN (FILESLOAD (FROM NOTECARDS)
NCTABLETOP)
(NCTableTop.NoteFileWhenOpenedFn NoteFile))))

(RPAQ? NCTableTop.MiddleButtonMenuItems

```
' ((Recompute% TableTop (FUNCTION NCTableTop.RecomputeTableTop)
  "Recomputes this table top using current configuration of its cards."
  (SUBITEMS (New% Cards (FUNCTION NCTableTop.RemakeTableTopSelectingNewCards)
    "Recompute this table top using a new list of cards."
    (|New Cards In Order| (FUNCTION NCTableTop.RemakeTableTopSelectingNewCardsInOrder)
      "Recompute this table top using a new list of cards, in the order in which they
      are selected.")
    (With% Template (FUNCTION NCTableTop.ComputeTableTopUsingTemplate)
      "Recompute this table top using a template table top.")))
  (|Add cards to TableTop| (FUNCTION NCTableTop.AddSelectedCardsToTableTop)
    "Add cards to the table top.")
  (|Remove cards from TableTop| (FUNCTION NCTableTop.RemoveSelectedCards)
    "Remove cards from the table top.")
  (|Show cards in TableTop| (FUNCTION NCTableTop.OpenCardInTableTop)
    "Bring up one of the cards in the table top choosing from a menu of card titles.")
  (|Bring up TableTop| (FUNCTION NCTableTop.BringUpTableTop)
    "Bring up the table top cards in proper layout.")
  (|Close down TableTop| (FUNCTION NCTableTop.CloseDownTableTop)
    "Close all cards in the table top.")))
```

;;; The following functions are exported, i.e. documented and callable from outside.

(DEFINEQ

(NCTableTop.BringUpTableTop

[LAMBDA (TableTopCardOrWindow DoNotUnCacheFlg) ; Edited 22-Sep-88 13:42 by jrc

```
;; Bring up all TableTopCard's cards in their proper positions.
;; rht 1/14/88: Now hangs the ShrunkenWinPos on the window around call to SHRINKW so that card window's SHRINKFN can find it.
;; rht 2/4/88: Moved guts of this function into NCTableTop.BringUpCardInTableTop.
;; rht 5/30/88: No longer uses NC.ActivateCardAndDo macro. We do it manually so can prevent uncaching in case where tabletop card is in its
;; own tabletop.
;; jrc 19-sep-88 -- returns the list of cards brought up, added DoNotUnCacheFlg param.
```

```
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
      WasCachedFlg CardsBroughtUp)
  (if (NOT (SETQ WasCachedFlg (NCP.CardCachedP TableTopCard)))
    then (NCP.CacheCards TableTopCard))
  (SETQ CardsBroughtUp (for TableTopEntry Card in (NCP.CardSubstance TableTopCard)
    when (SETQ Card (NCTableTop.BringUpCardInTableTop TableTopEntry)) collect Card))
  (if (NOT (OR WasCachedFlg DoNotUnCacheFlg (NCP.CardWindow TableTopCard)))
    then (NCP.CloseCards TableTopCard))
  CardsBroughtUp])
```

(NCTableTop.OpenCardInTableTop

[LAMBDA (TableTopCardOrWindow InterestedWindow) ; Edited 30-May-88 15:14 by Trigg

;;; Open one of the cards in TableTopCard.

;; pmi 2/18/88: Added check for non-nil menu items, in case the TableTop card has no entries.
;; rht 5/30/88: No longer uses NC.ActivateCardAndDo macro. We do it manually so can prevent uncaching in case where tabletop card itself was
;; the one brought up.

```
(DECLARE (GLOBALVARS NC.MsgDelay))
(OR InterestedWindow (SETQ InterestedWindow (NCP.CoerceToInterestedWindow TableTopCardOrWindow)))
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
      WasCachedFlg MenuItems Menu)
      (if (NOT (SETQ WasCachedFlg (NCP.CardCachedP TableTopCard)))
          then (NCP.CacheCards TableTopCard))
      (if [SETQ MenuItems (for TableTopEntry in (NCP.CardSubstance TableTopCard) bind Card
          when (NCP.ValidCardP (SETQ Card (fetch (TableTopEntry Card) of TableTopEntry)))
          collect ` (, (NCP.CardTitle Card)
                    ', TableTopEntry "Open Card having given title.")]
          then (SETQ Menu (create MENU
                                ITEMS _ MenuItems
                                TITLE _ "Open card in TableTop?"))
              (NCTableTop.BringUpCardInTableTop (MENU Menu))
          else (NC.PrintMsg InterestedWindow T "No cards in this TableTop.")
              (DISMISS NC.MsgDelay)
              (NC.ClearMsg InterestedWindow T))
          (if (AND (NOT WasCachedFlg)
                  (NOT (NCP.CardWindow TableTopCard)))
              then (NCP.CloseCards TableTopCard]))
```

(NCTableTop.CloseDownTableTop

[LAMBDA (TableTopCardOrWindow) ; Edited 5-Apr-89 15:25 by pmi

;; Close all the cards in the table top.
;; pmi 4/4/89: Now passes T to NCP.CloseCards as Don'tForceFilingFlg (new). Keeps cards in tabletops from bugging user about filing when the
;; tabletop is closed down.

```
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
      (NC.ActivateCardAndDo TableTopCard (for TableTopEntry in (NCP.CardSubstance TableTopCard) bind Card
          when (NCP.ValidCardP (SETQ Card (fetch (TableTopEntry Card)
          of TableTopEntry)))
          do (NCP.CloseCards Card T ]))
```

(NCTableTop.RemakeTableTop

[LAMBDA (TableTopCardOrWindow ListOfCards RetainClosedCardsFlg NoToTopOrder) ; Edited 4-May-89 10:18 by pmi

;; Rebuild the substance of TableTopCard using a new list of cards.
;; pmi 2/19/88: Added RetainClosedCardsFlg argument to be passed in as T when we know that we want closed cards included in the TableTop.
;; Also added check for 'CANCELLED'.
;; pmi 5/4/89: Added NoToTopOrder argument. If non-NIL, disables ordering of cards in tabletop by their TOTOPW order. The order in which the
;; cards are selected is used instead.

```
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
      NewTableTop)
      (OR (NC.ActiveCardP TableTopCard)
          (NCP.CacheCards TableTopCard))
      (if (NEQ (SETQ NewTableTop (NCTableTop.ComputeTableTop ListOfCards TableTopCard RetainClosedCardsFlg
          NoToTopOrder))
          'CANCELLED)
          then (NC.SetSubstance TableTopCard NewTableTop)
              (NC.MarkCardDirty TableTopCard]))
```

(NCTableTop.RecomputeTableTop

[LAMBDA (TableTopCardOrWindow NoToTopOrder) ; Edited 4-May-89 10:20 by pmi

;; Reusing the cards in the TableTopCard, recompute the substance based on the card's new shapes, positions, and totopw ordering.
;; pmi 2/19/88: Added check for 'CANCELLED'.
;; pmi 5/4/89: Added NoToTopOrder argument. If non-NIL, disables ordering of cards in tabletop by their TOTOPW order. The order in which the
;; cards are selected for the tabletop is used instead.

```
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
      NewTableTop)
      (OR (NC.ActiveCardP TableTopCard)
          (NCP.CacheCards TableTopCard))
      (if (NEQ (SETQ NewTableTop (NCTableTop.ComputeTableTop (for TableTopEntry in (NC.FetchSubstance
          TableTopCard)
          bind Card
          when (NCP.ValidCardP (SETQ Card
          (fetch (TableTopEntry
          Card)
          of TableTopEntry)))
          collect Card)
          TableTopCard NIL NoToTopOrder))
          'CANCELLED)
          then (NC.SetSubstance TableTopCard NewTableTop)
              (NC.MarkCardDirty TableTopCard]))
```

(NCTableTop.RemoveCardsFromTableTop

```
[LAMBDA (TableTopCardOrWindow CardsToRemove) ; Edited 27-Jun-88 11:26 by Trigg
;; Remove the cards in CardsToRemove from the tabletop.
;; rht&jrc 6/27/88: Now returns the cards actually removed.
(if CardsToRemove
  then (LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
             NewSubstance InitialSubstance)
        (OR (NCP.CardCachedP TableTopCard)
            (NCP.CacheCards TableTopCard))
        (SETQ InitialSubstance (NCP.CardSubstance TableTopCard))
        (SETQ NewSubstance (for TableTopEntry in InitialSubstance
                               unless (FMEMB (fetch (TableTopEntry Card) of TableTopEntry)
                                           CardsToRemove)
                                       collect TableTopEntry))
        (if (LESSP (LENGTH NewSubstance)
                  (LENGTH InitialSubstance))
            then (NCP.CardSubstance TableTopCard NewSubstance)
              (NCP.MarkCardDirty TableTopCard))
        ;; Return the cards that were actually removed.
        (for TableTopEntry in InitialSubstance bind Card when (FMEMB (SETQ Card (fetch (TableTopEntry
                                                                                          Card)
                                                                                          of TableTopEntry))
                                                                      CardsToRemove)
                    collect Card])
```

(NCTableTop.AddCardsToTableTop

```
[LAMBDA (TableTopCardOrWindow CardsToAdd InterestedWindow) ; Edited 27-Jun-88 11:35 by Trigg
;; Add the cards in CardsToAdd to the tabletop.
;; rht&jrc 6/27/88: Now returns any new cards added.
(DECLARE (GLOBALVARS NC.MsgDelay))
(OR InterestedWindow (SETQ InterestedWindow (NCP.CoerceToInterestedWindow TableTopCardOrWindow)))
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
      OldTableTopCards TotalCards NewTableTop)
  (SETQ OldTableTopCards (NCTableTop.CardsInTableTop TableTopCard))
  (if (NEQ (SETQ TotalCards (UNION CardsToAdd OldTableTopCards))
        OldTableTopCards)
      then (OR (NC.ActiveCardP TableTopCard)
               (NCP.CacheCards TableTopCard))
          (if (NEQ (SETQ NewTableTop (NCTableTop.ComputeTableTop TotalCards TableTopCard T))
                  'CANCELLED)
              then (NC.SetSubstance TableTopCard NewTableTop)
                (NC.MarkCardDirty TableTopCard)
              ;; If successful, return new cards added.
              (LDIFFERENCE TotalCards OldTableTopCards)
            else NIL)
      else (NCP.PrintMsg InterestedWindow T "No new cards specified.")
           (DISMISS NC.MsgDelay)
           (NC.ClearMsg InterestedWindow T)
           NIL))
```

(DEFINEQ

(NCTableTop.CardsInTableTop

```
[LAMBDA (TableTopCard) ; Edited 11-May-88 11:27 by Trigg
;; Return the set of cards appearing in this tabletop.
(NC.ActivateCardAndDo TableTopCard (for TableTopEntry in (NCP.CardSubstance TableTopCard) bind Card
                                     when (NCP.ValidCardP (SETQ Card (fetch (TableTopEntry Card)
                                                                              of TableTopEntry)))
                                     collect Card])
```

(NCTableTop.CollectCards

```
[LAMBDA (RootTableTopCards LinkTypes MaxDepth FollowCrossFileLinksFlg) ; Edited 11-May-88 11:32 by Trigg
;; Return the list of cards reachable from the given tabletops following given link types up to given MaxDepth.
(SETQ RootTableTopCards (MKLIST RootTableTopCards))
(if (GEQ MaxDepth 1)
  then (UNION RootTableTopCards (NCP.CollectCards (for TableTopCard in RootTableTopCards
                                                    join (NCTableTop.CardsInTableTop TableTopCard))
                                             LinkTypes
                                             (SUB1 MaxDepth)
                                             FollowCrossFileLinksFlg))
  else RootTableTopCards])
```

)

(DEFINEQ

(NCTableTop.TableTopBasedP

[LAMBDA (CardOrCardType)

; Edited 12-Jul-88 18:14 by Trigg

;; Return non-nil if CardOrCardType is a subtype of the TableTop card type. If CardOrCardType is a card, use its card type.

(NCP.IsSubTypeOfP (if (NC.CardP CardOrCardType)
then (NCP.CardType CardOrCardType)
else CardOrCardType)
' TableTop])

)

;;; The rest are internal.

(DEFINEQ

(NCTableTop.ComputeTableTop

[LAMBDA (ListOfCards TableTopCard RetainClosedCardsFlg NoToTopOrder)

; Edited 22-Aug-89 19:01 by pmi

;; This looks at all the open windows and computes the tabletop, a list of records containing card, windowregion, shrunken window position, and
;; flag indicating whether window is shrunken. Only done for those entries in ListOfCards that have open windows. TOTOPW order is preserved.

;; rht 1/21/88: Had to do REVERSE to get the right to-top order. Wonder whether (OPENWINDOWS) reversed in lyric.

;; pmi 2/17/88: Added check for ListOfCards in case user cancels operation.

;; pmi 2/19/88: Added TableTopCard and RetainClosedCardsFlg arguments. If RetainClosedCardsFlg is non-NIL, cards that are closed will be
;; added to the TableTop using their region stored in the notefile. If RetainClosedCardsFlg is NIL, the user will be asked to confirm deleting closed
;; cards from the TableTop.

;; rht 5/13/88: Simplified code for above fix a bit and made sure all the variables were bound. Also added call to

;; NCTableTop.ComputeTTEnterProps.

;; pmi 5/4/89: TOTOPW order is preserved unless NoToTopOrder is non-NIL. In that case, keeps entries in tabletop in same order as in

;; ListOfCards, so user can control order in which cards are opened.

(RESETLST

(RESETSAVE (CURSOR WAITINGCURSOR))

[if (NEQ ListOfCards 'DON'T)

then (LET (OpenWindows NewTableTopEntries OpenCardsList ClosedCardsList ClosedCardTTEnterPairs

AdditionalTableTopEntries)

(SETQ OpenWindows (OPENWINDOWS))

[if NoToTopOrder

then [SETQ NewTableTopEntries (for Card in ListOfCards bind Win ShrunkenFlg Icon IconWin

when (AND (SETQ Win (NCP.CardWindow Card))

(OR (SETQ ShrunkenFlg

(FMEMB (SETQ Icon (WINDOWPROP

Win

' ICON))

OpenWindows))

(FMEMB Win OpenWindows)))

collect (push OpenCardsList Card)

(create TableTopEntry

Card _ Card

Region _ (WINDOWPROP Win 'REGION)

ShrunkenWinPos _

(COND

(Icon (WINDOWPOSITION Icon))

([WINDOWP (SETQ IconWin

(WINDOWPROP Win

' ICONWINDOW]

(WINDOWPOSITION IconWin)))

ShrunkenFlg _ (AND ShrunkenFlg T)

Props _

(if ShrunkenFlg

then (NCTableTop.ComputeTTEnterProps

Card Icon Win)

else (NCTableTop.ComputeTTEnterProps

Card Win]

else (SETQ NewTableTopEntries

(REVERSE (for Win in OpenWindows bind Card IconFor IconWin

when (AND (SETQ Card (NC.CoerceToCard (OR (SETQ IconFor

(WINDOWPROP Win 'ICONFOR))

Win)))

(FMEMB Card ListOfCards))

collect (push OpenCardsList Card)

(create TableTopEntry

Card _ Card

Region _ (WINDOWPROP (OR IconFor Win)

' REGION)

ShrunkenWinPos _ (COND

(IconFor (WINDOWPOSITION Win))

([WINDOWP (SETQ IconWin

(WINDOWPROP Win

' ICONWINDOW]

(WINDOWPOSITION IconWin)))

ShrunkenFlg _ (AND IconFor T)

Props _ (NCTableTop.ComputeTTEnterProps Card Win IconFor]

```

;; Get the poop on cards in this tabletop that are currently closed.
(SETQ ClosedCardsList (LDIFFERENCE ListOfCards OpenCardsList))
(SETQ ClosedCardTTEEntryPairs (for TableTopEntry in (NCP.CardSubstance TableTopCard)
                                bind Card when (AND (NCP.ValidCardP (SETQ Card
                                                                    (fetch (TableTopEntry
                                                                    Card)
                                                                    of TableTopEntry)))
                                                (FMEMB Card ClosedCardsList))
                                collect (CONS Card TableTopEntry)))
(if RetainClosedCardsFlg
    then ;; If some of the cards are closed, use any saved version or make a new one.
        (NCONC (for ClosedCard in ClosedCardsList when (NCP.ValidCardP ClosedCard)
                collect (OR (CDR (FASSOC ClosedCard ClosedCardTTEEntryPairs))
                            (create TableTopEntry
                                    Card _ ClosedCard
                                    Region _ (NCP.CardRegion ClosedCard)
                                    ShrunkenFlg _ NIL)))
                NewTableTopEntries)
    else ;; If some of the cards are closed, check with the user about deleting them from the TableTop.
        [SETQ AdditionalTableTopEntries
         (for ClosedCard in ClosedCardsList
          when [AND (NCP.ValidCardP ClosedCard)
                  (SELECTQ (RESETLIST
                            (RESETSAVE (CURSOR T))
                            (NC.AskUserWithMenu ' (Cancel Keep Remove)
                                                (MKSTRING (CONCAT "Card %" (NC.FetchTitle
                                                                    ClosedCard)
                                                                    "%" is closed. Keep it or remove
                                                                    from this tabletop?"))
                            (NCP.CoerceToInterestedWindow TableTopCard)))
                  (Cancel (RETURN 'CANCELLED))
                  (Keep T)
                  (Remove NIL)
                  (RETURN 'CANCELLED)]
          collect (OR (CDR (FASSOC ClosedCard ClosedCardTTEEntryPairs))
                    (create TableTopEntry
                            Card _ ClosedCard
                            Region _ (NCP.CardRegion ClosedCard)
                            ShrunkenFlg _ NIL))
          (if (EQ AdditionalTableTopEntries 'CANCELLED)
              then 'CANCELLED
              else (NCONC AdditionalTableTopEntries NewTableTopEntries)))]

```

(NCTableTop.ComputeTTEEntryProps

[LAMBDA (Card Window IconFor)

; Edited 22-Aug-89 19:01 by pmi

;; Extract info from Card and its window about its display. This includes scrolling info as well as other cardtype-dependent junk. If Card is
 ;; shrunken, then Window is the shrunken icon for IconFor window. Else IconFor should be nil.
 ;; rht 7/21/88: Now computes scrolling info for File cards as well.
 ;; pmi 3/28/89: Now Window will always be expanded version, not shrunken. (See changes to NCTabletop.ComputeTableTop). Removed IconFor
 ;; Argument.
 ;; pmi 8/22/89: No, Window will not always be expanded, so reinstated IconFor Argument.

```

(COND
  [(OR (NCP.TextBasedP Card)
        (NCP.IsSubTypeOfP (NCP.CardType Card)
                          'File))
   '(ScrollingInfo ,(NC.UpperLeftCH# (OR IconFor Window))]
  [(NCP.GraphBasedP Card)
   '(ScrollingInfo ,(DSPCLIPPINGREGION NIL Window)
                   ,@[if (FMEMB (WINDOWPROP Window 'BrowserOverviewWin)
                               (ATTACHEDWINDOWS Window))
                       then '(BrowserOverviewSpecs ,(create NCTableTop.BrowserOverviewSpecs
                                                            OverviewWinWidth _ (WINDOWPROP Window
                                                                    'OverviewWinWidth)
                                                            OverviewWinHeight _ (WINDOWPROP Window
                                                                    'OverviewWinHeight)
                                                            WHERETOATTACHOVERVIEWWIN _ (WINDOWPROP Window
                                                                    'WHERETOATTACHOVERVIEWWIN))
                           ,
                           WHERETOATTACHOVERVIEWWIN
                           )
                   OVERVIEWWINMODE _ (WINDOWPROP Window 'OVERVIEWWINMODE)]
   ,@(if (for Win in (ATTACHEDWINDOWS Window) never (WINDOWPROP Win 'LINKSLEGENDWINP))
         then '(ClosedLinksLegendFlg T))
   ,@(if (for Win in (ATTACHEDWINDOWS Window) thereis (WINDOWPROP Win 'GRAPHEDITMENUFLG))
         then '(GraphEditMenuFlg T)]
  ((NCP.SketchBasedP Card)
   '(ScrollingInfo ,(DSPCLIPPINGREGION NIL Window)
                   ,@[if [NOT (OPENWP (WINDOWPROP Window 'SKETCHFIXEDMENU)
                                       then '(ClosedSketchMenuFlg T)]

```

(NCTableTop.ComputeCardsFromTemplate

```

[LAMBDA (TableTopCard Window TemplateCard) ; Edited 3-Apr-89 17:05 by pmi
;; If TemplateCard is NIL, asks user to pick one from existing table tops. Then asks user to select cards to persist and cards to copy for the new
;; table top.
(LET
  ((InterestedWindow (OR (OPENWP Window)
                        (NCP.CoerceToInterestedWindow TableTopCard)))
   CardsInTemplate PersistentCards CardsToBeCopied)
  (OR
   TemplateCard
   (if (EQ (SETQ TemplateCard (NCP.SelectCards InterestedWindow T (FUNCTION NCP.ValidCardP)
                                             "Select a table top card to use as a template for the new table top." T))
       'DON'T)
       else (SETQ CardsInTemplate (NCTableTop.CardsInTableTop TemplateCard)
              (NCTableTop.BringUpTableTop TemplateCard)
              (if (EQ (SETQ PersistentCards (NCP.SelectCards InterestedWindow NIL '[LAMBDA (Card)
                                                                    (MEMB Card CardsInTemplate)]
                                                                    "Select cards to persist from this table top." T))
                  'DON'T)
                  elseif (EQ (SETQ CardsToBeCopied (NCP.SelectCards InterestedWindow NIL '[LAMBDA (Card)
                                                                    (MEMB Card
                                                                    CardsInTemplate]
                                                                    "Select cards to be copied from this table top." T))
                              'DON'T)
                              else (PROG1 [for OldCard in CardsInTemplate bind NewCard NewCardWindow NewCardTextStream
                                                                    NewCardTextLength OldCardIconWindow
                                                                    OldCardIconWindowRegion ShrunkenWinPos
                                                                    when (OR (FMEMB OldCard PersistentCards)
                                                                    (FMEMB OldCard CardsToBeCopied))
                                                                    collect (if (FMEMB OldCard PersistentCards)
                                                                    then OldCard
                                                                    else (PROG1 (SETQ NewCard (CAR (NCP.CopyCards OldCard NIL NIL T 'DON'TCOPY
                                                                    InterestedWindow T)))
                                                                    (SETQ NewCardWindow (NCP.OpenCard NewCard (NCP.CardRegion OldCard)
                                                                    )
                                                                    )
                                                                    (SETQ NewCardTextStream (TEXTSTREAM NewCardWindow))
                                                                    (SETQ NewCardTextLength (TEDIT.NCHARS NewCardTextStream))
                                                                    ;; Delete any text in the copy, hopefully preserving the looks for new text.
                                                                    (if (GREATERP NewCardTextLength 0)
                                                                    then (TEDIT.DELETE NewCardTextStream 1 NewCardTextLength))
                                                                    (if [OPENWP (SETQ OldCardIconWindow (WINDOWPROP (NCP.CardWindow
                                                                    OldCard)
                                                                    'ICON]
                                                                    then (SETQ OldCardIconWindowRegion (WINDOWPROP
                                                                    OldCardIconWindow
                                                                    'REGION))
                                                                    (SETQ ShrunkenWinPos (create POSITION
                                                                    XCOORD _ (fetch (REGION LEFT)
                                                                    of
                                                                    OldCardIconWindowRegion
                                                                    )
                                                                    YCOORD _ (fetch (REGION BOTTOM)
                                                                    of
                                                                    OldCardIconWindowRegion
                                                                    )))
                                                                    (WINDOWPROP NewCardWindow 'SHRUNKENWINPOS ShrunkenWinPos)
                                                                    (SHRINKW NewCardWindow NIL ShrunkenWinPos)
                                                                    (WINDOWPROP NewCardWindow 'SHRUNKENWINPOS NIL))])
                                                                    (if PersistentCards
                                                                    then (NCP.CloseCards (LDIFFERENCE CardsInTemplate PersistentCards)
                                                                    T)
                                                                    else (NCP.CloseCards CardsInTemplate T))))))

```

(NCTableTop.BringUpCardInTableTop

[LAMBDA (TableTopEntry) ; Edited 19-Sep-88 16:30 by jrc

;; Bring up particular card in tabletop designated by TableTopEntry.
;; rht 5/13/88: Now calls NCTableTop.ProcessTTEnterProps to do things like scroll window properly.
;; jrc 19-sep-88 returns card, if it was opened.

```

(LET ((Card (fetch (TableTopEntry Card) of TableTopEntry))
      (Region (fetch (TableTopEntry Region) of TableTopEntry))
      (ShrunkenWinPos (fetch (TableTopEntry ShrunkenWinPos) of TableTopEntry))
      Win)
  (if (AND (NCP.ValidCardP Card)
          (REGIONP Region))
      then (SETQ Win (NCP.OpenCard Card Region))
           (NCTableTop.ProcessTTEnterProps Card Win (fetch (TableTopEntry Props) of TableTopEntry))
           (if (AND (fetch (TableTopEntry ShrunkenFlg) of TableTopEntry)
                   (POSITIONP ShrunkenWinPos))
               then (WINDOWPROP Win 'SHRUNKENWINPOS ShrunkenWinPos)
                   (SHRINKW Win NIL ShrunkenWinPos)
                   (WINDOWPROP Win 'SHRUNKENWINPOS NIL))
           Card])

```

(NCTableTop.ProcessTTEnterProps

```

[LAMBDA (Card Window Props) ; Edited 21-Jul-88 21:07 by Trigg
;; Massage Window according to Props including scrolling.
;; rht 7/21/88: Now processes scrolling info for File cards as well.
(LET [(ScrollingInfo (LISTGET Props 'ScrollingInfo)]
;; Handle scrolling first
[COND
((OR (NCP.TextBasedP Card)
(NCP.IsSubTypeOfP (NCP.CardType Card)
'File))
(if (AND (NUMBERP ScrollingInfo)
(GEQ ScrollingInfo 2))
then (NC.ScrollToCH# Window ScrollingInfo)))
((REGIONP ScrollingInfo)
(LET ((CurDisplayRegion (DSPCLIPPINGREGION NIL Window))
(SCROLLW Window (DIFFERENCE (fetch (REGION LEFT) of CurDisplayRegion)
(fetch (REGION LEFT) of ScrollingInfo))
(DIFFERENCE (fetch (REGION BOTTOM) of CurDisplayRegion)
(fetch (REGION BOTTOM) of ScrollingInfo)
;; Take care of random browser and sketch processing
(COND
((NCP.GraphBasedP Card)
;; If there are saved specs for the browser overview, then install them on the window and make a browser overview win.
(LET [(BrowserOverviewSpecs (LISTGET Props 'BrowserOverviewSpecs)]
(if BrowserOverviewSpecs
then (WINDOWPROP Window 'OverviewWinWidth (fetch (NCTableTop.BrowserOverviewSpecs
OverviewWinWidth)
of BrowserOverviewSpecs))
(WINDOWPROP Window 'OverviewWinHeight (fetch (NCTableTop.BrowserOverviewSpecs
OverviewWinHeight)
of BrowserOverviewSpecs))
(WINDOWPROP Window 'WHEREOATTACHOVERVIEWWIN (fetch (NCTableTop.BrowserOverviewSpecs
WHEREOATTACHOVERVIEWWIN)
of BrowserOverviewSpecs))
(WINDOWPROP Window 'OVERVIEWWINMODE (fetch (NCTableTop.BrowserOverviewSpecs
OVERVIEWWINMODE)
of BrowserOverviewSpecs))
(NC.MakeBrowserOverviewWin Window)))
;; Check whether links legend window should be closed..
(if (LISTGET Props 'ClosedLinksLegendFlg)
then (for Win in (ATTACHEDWINDOWS Window) when (WINDOWPROP Win 'LINKSLEGENDWINP)
do (CLOSEW Win)))
;; Check whether graph edit menu needs to be attached.
(if (LISTGET Props 'GraphEditMenuFlg)
then (NC.SetUpGraphEditMenus Window)
(NC.BrowserFixGraphEditMenuFn Window)))
((NCP.SketchBasedP Card)
;; Check whether sketch menu should be closed..
(if (LISTGET Props 'ClosedSketchMenuFlg)
then (LET [(SketchMenuWin (WINDOWPROP Window 'SKETCHFIXEDMENU)
(AND (OPENWP SketchMenuWin)
(CLOSEW SketchMenuWin)
)
)
(DEFINEQ

```

(NCTableTop.RemakeTableTopSelectingNewCards

```

[LAMBDA (TableTopCardOrWindow InterestedWindow NoToTopOrder) ; Edited 4-May-89 14:15 by pmi
;;; Rebuild the substance of TableTopCard using a new list of cards provided by the user.
;; rht 1/23/88: Added InterestedWindow and pass to NCP.SelectCards.
;; pmi 2/19/88: Now passes T to NCTableTop.RemakeTableTop as RetainClosedCardsFlg - this allows the user to select new cards from link icons
;; even though they are not up on the screen.
;; pmi 5/4/89: Added NoToTopOrder argument. If non-NIL, disables ordering of cards in tabletop by their TOTOPW order. The order in which the
;; cards are selected is used instead.
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
ListOfCards)
(SETQ ListOfCards (NCP.SelectCards (OR (OPENWP InterestedWindow)
TableTopCard)
NIL
(FUNCTION NCP.ValidCardP)
"Choose new cards for rebuilding table top." T))
(if (EQ ListOfCards 'DON'T)
then NIL
else (NCTableTop.RemakeTableTop TableTopCard ListOfCards T NoToTopOrder]))

```


(NCTableTop.RemakeTableTopSelectingNewCardsInOrder

```
[LAMBDA (TableTopCardOrWindow InterestedWindow) ; Edited 4-May-89 14:22 by pmi
;; Remake the tabletop using the order of selection for the order of the cards in the tabletop.
(NCTableTop.RemakeTableTopSelectingNewCards TableTopCardOrWindow InterestedWindow T)]
```

(NCTableTop.ComputeTableTopUsingTemplate

```
[LAMBDA (TableTopCardOrWindow InterestedWindow) ; Edited 4-May-89 15:19 by pmi
```

;;; Rebuild the substance of TableTopCard using a template card provided by the user.

```
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
ListOfCards)
(SETQ ListOfCards (NCTableTop.ComputeCardsFromTemplate TableTopCard InterestedWindow))
(if (EQ ListOfCards 'DON'T)
then NIL
else (NCTableTop.RemakeTableTop TableTopCard ListOfCards T]))
```

(NCTableTop.RemoveSelectedCards

```
[LAMBDA (TableTopCardOrWindow InterestedWindow) ; Edited 27-Jun-88 11:26 by Trigg
```

;;; User selects a set of cards to remove from the tabletop. User picks cards to remove by successive selections from an ever shrinking menu of card titles.

;;; 16-jun-88 jrc. Function now returns the list of cards removed, or NIL if none removed.

```
(DECLARE (GLOBALVARS NC.MsgDelay))
(OR InterestedWindow (SETQ InterestedWindow (NCP.CoerceToInterestedWindow TableTopCardOrWindow)))
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
CardsToRemove MenuItems)
(if [SETQ MenuItems (for Card in (NCTableTop.CardsInTableTop TableTopCard)
collect ` (, (NCP.CardTitle Card)
', Card "Remove card with this title from tabletop."
then (SETQ CardsToRemove (for bind ChosenCard
eachtime [SETQ ChosenCard
(AND MenuItems
(MENU (create MENU
ITEMS _ MenuItems
TITLE _ "Choose cards to remove"]
while ChosenCard collect ; Shrink the MenuItems by one item and ask again.
(SETQ MenuItems
(DREMOVE (for MenuItem in MenuItems
thereis (EQ ChosenCard (CADADR
MenuItem
MenuItems))
ChosenCard))
(if (AND CardsToRemove (NCP.AskYesOrNo (CONCAT "You've selected " (LENGTH CardsToRemove)
" card(s) to remove."
(Character 13))
"Okay to remove them? " "Yes" T InterestedWindow))
then (NCTableTop.RemoveCardsFromTableTop TableTopCard CardsToRemove))
else (NCP.PrintMsg InterestedWindow T "No cards in this tabletop.")
(DISSMISS NC.MsgDelay)
(NC.ClearMsg InterestedWindow T)
NIL])
```

(NCTableTop.AddSelectedCardsToTableTop

```
[LAMBDA (TableTopCardOrWindow InterestedWindow) ; Edited 27-Jun-88 11:35 by Trigg
```

;;; Add some new cards provided by the user to the tabletop.

```
(LET ((TableTopCard (NCP.CoerceToCard TableTopCardOrWindow))
CardsToAdd)
(SETQ CardsToAdd (NCP.SelectCards (OR (OPENWP InterestedWindow)
TableTopCard)
NIL
(FUNCTION NCP.ValidCardP)
"Choose cards to add to table top." T))
(if (EQ CardsToAdd 'DON'T)
then NIL
else (NCTableTop.AddCardsToTableTop TableTopCard CardsToAdd InterestedWindow))
```

)

(DEFINEQ

(NCTableTop.AddOpenEventsExpr

```
[LAMBDA (NoteFile) (* rht%: "16-May-87 20:12")
```

(* Add an expression to the open events card for NoteFile, if it's not already there.)

(* rht 4/17/87%: Changed from FILESLOAD to call NC.LoadFileFromDirectories so that TTY window gets handled same as rest of NoteCards.)


```

(|Close Saving TableTop|
NCTableTop.CloseNoteFileSavingTableTop
"Close this notefile, saving
currently open cards in a
tabletop.")]
(NoteFileOperationsCloseItem '(| Close | (NC.DoNoteFileOp 'Close% NoteFile)
"Close a notefile."
(SUBITEMS (|Close w/o confirm| (NC.DoNoteFileOp
'|Close w/o confirm|)
"Close a notefile w/o asking for user confirm
before closing open cards.")
(Close% All (NC.CloseSession)
"Close all currently open NoteFiles")
(|Close All w/o confirm| (NC.CloseSession NIL T)
"Close all open notefiles w/o asking for user
confirm before closing open cards.")
(|Close Saving TableTop| (NC.DoNoteFileOp
(FUNCTION
NCTableTop.CloseNoteFileSavingTableTop
))
"Close a notefile, saving currently open cards in a
tabletop.")
(|Close All Saving TableTops| (
NCTableTop.CloseSessionSavingTableTops
)
"Close all open notefiles saving their tabletops.")]
(PUTASSOC 'Close (CDR NoteFileIconCloseItem)
NC.NoteFileIconOperationsMenuItems)
(pushnew NC.NoteFileIconOpenOperations '|Close Saving TableTop|)
(NC.MakeNoteFileIconOperationsMenus)
(PUTASSOC '| Close | (CDR NoteFileOperationsCloseItem)
NC.NoteFileOpsItems)
(SETQ NC.NoteFileOpsMenu NIL])

```

(NCTableTop.CloseNoteFileSavingTableTop

[LAMBDA (NoteFile NoteFileIconWindow) (* rht%: "30-Mar-87 23:31")

(* Close down the notefile, but save table top first.)
(* rht 3/30/87%: Minor change to keep it from bugging about saving open cards.)

```

(NCTableTop.MakeOrUpdateNoteFileTableTop NoteFile)
(NCTableTop.AddOpenEventsExpr NoteFile)
(NC.CloseNoteFile NoteFile NoteFileIconWindow NIL T))

```

(NCTableTop.CloseSessionSavingTableTops

[LAMBDA NIL ; Edited 8-Mar-88 14:37 by pmi

(* Close all open notefiles saving their table tops.)

```

(DECLARE (GLOBALVARS NC.NoteFilesHashArray))
[MAPHASH NC.NoteFilesHashArray (FUNCTION (LAMBDA (Value Key)
(if (AND (type? NoteFile Value)
(NC.NoteFileOpenP Value))
then (NCTableTop.CloseNoteFileSavingTableTop Value)
))
(NC.PrintMsg NIL T "Done closing all NoteFiles.")]
)

```

(DEFINEQ

(NCTableTop.MakeFn

[LAMBDA (TableTopCard Title NoDisplayFlg ListOfCards InterestedWindow RegionOrPosition) ; Edited 29-Mar-89 16:23 by pmi

:: Create a table top card substance. If ListOfCards is nil, ask user to select cards. Substance consists of list of pairs of card uid and screen
:: window region. Note that OPENWINDOWS is used to get proper TOTOPW ordering or cards.
:: rht 1/14/88: Now passes TableTopCard as instigating card to NCP.SelectCards in case NoDisplayFlg is set.
:: rht 1/23/88: Incredible kludge: NoDisplayFlg can be a window in which case pass it to NCP.SelectCards.
:: pmi 2/19/88: Added TableTopCard and T to call to NCTableTop.ComputeTableTop. T is the RetainClosedCardsFlg, and indicates that some of
:: ListOfCards may be closed, and should be included in the TableTop.
:: pmi 2/26/88: Solved NoDisplayFly kludge by passing in InterestedWindow argument.
:: rht 8/5/88: Added RegionOrPosition arg and passed to NC.MakeNewCardWindow.

```

(LET (Window)
(if (NOT NoDisplayFlg)
then (SETQ Window (NC.MakeNewCardWindow TableTopCard (OR Title "Untitled")
NIL NIL RegionOrPosition))
(NCP.SetUpTitleBar Window (NC.RetrieveType TableTopCard))
(WINDOWADDPROP Window 'SHRINKFN (FUNCTION (NC.GraphCardShrinkFn))
else (SETQ Window InterestedWindow))
[OR (LISTP ListOfCards)
(if (NCP.AskYesOrNo "Use template table top? " NIL NIL T Window)
then (SETQ ListOfCards (NCTableTop.ComputeCardsFromTemplate TableTopCard Window))

```

```

else (SETQ ListOfCards (NCP.SelectCards (OR (OPENWP Window)
                                           TableTopCard)
                                           NIL
                                           (FUNCTION NCP.ValidCardP)
                                           "Choose cards for table top." T]
      (if (EQ ListOfCards 'DON'T)
          then (NC.DeactivateCard TableTopCard)
              (AND (WINDOWP Window)
                   (CLOSEW Window))
              NIL
          else (NC.SetSubstance TableTopCard (NCTableTop.ComputeTableTop ListOfCards TableTopCard T)
              (if NoDisplayFlg
                  then TableTopCard
                  else Window]))

```

(NCTableTop.EditFn

[LAMBDA (TableTopCard Substance Region/Position) ; Edited 30-May-88 15:05 by Trigg

;; Bringing up a table top card, for now, means to bring up all its cards in their proper positions.
 ;; rht 1/20/88: For some reason, we were telling NC.MakeNewCardWindow not to bring up the window. No more.
 ;; rht 5/30/88: Now passes Region/Position to NC.MakeNewCardWindow.

```

(LET (Window)
    (if (SETQ Window (NC.FetchWindow TableTopCard))
        then (TOPOW Window)
             (if Region/Position
                 then (SHAPEW Window (NC.DetermineDisplayRegion TableTopCard Region/Position)))
             (RPTQ 2 (FLASHW Window))
        else (SETQ Window (NC.MakeNewCardWindow TableTopCard (OR (NC.RetrieveTitle TableTopCard)
                                                                "Untitled")
                                                                NIL NIL Region/Position))
            (NCP.SetUpTitleBar Window (NC.RetrieveType TableTopCard))
            (WINDOWADDPROP Window 'SHRINKFN (FUNCTION NC.GraphCardShrinkFn)))
    Window])

```

(NCTableTop.PutFn

[LAMBDA (TableTopCard Stream) ; Edited 22-Jul-88 00:06 by Trigg

;; Write the table top substance down to the stream. The format is number of pairs, followed by pairs of uid and region.

;; rht 5/13/88: Now writes down new Props field of TableTopEntry datatype.
 ;; rht 7/22/88: Now writes down notefile uid for cards in tabletopentries.

```

(LET ((NoteFile (NCP.CardNoteFile TableTopCard))
      (Substance (NCP.CardSubstance TableTopCard))
      (StartLoc (GETFILEPTR Stream))
      (NumRecords 0))
    (NC.WritePtr Stream 0 3)
    [for TableTopEntry in Substance
        do (LET ((Card (fetch (TableTopEntry Card) of TableTopEntry))
                (Region (fetch (TableTopEntry Region) of TableTopEntry))
                (ShrunkenWinPos (fetch (TableTopEntry ShrunkenWinPos) of TableTopEntry))
                (ShrunkenFlg (fetch (TableTopEntry ShrunkenFlg) of TableTopEntry))
                Card)
            (if (AND (NCP.ValidCardP Card)
                    (REGIONP Region)
                    (OR (NULL ShrunkenFlg)
                       (POSITIONP ShrunkenWinPos)))
                then (NC.WriteUID Stream (fetch (Card UID) of Card))
                    (NC.WriteUID Stream (fetch (NoteFile UID) of (NCP.CardNoteFile Card)))
                    (NC.WritePtr Stream (fetch LEFT of Region)
                                         2)
                    (NC.WritePtr Stream (fetch BOTTOM of Region)
                                         2)
                    (NC.WritePtr Stream (fetch WIDTH of Region)
                                         2)
                    (NC.WritePtr Stream (fetch HEIGHT of Region)
                                         2)
                    (NC.WritePtr Stream (if ShrunkenFlg
                                            then (fetch XCOORD of ShrunkenWinPos)
                                            else 0)
                                         2)
                    (NC.WritePtr Stream (if ShrunkenFlg
                                            then (fetch YCOORD of ShrunkenWinPos)
                                            else 0)
                                         2)
                    (NC.WritePtr Stream (if ShrunkenFlg
                                            then 1
                                            else 0)
                                         1)
                    (NC.WritePropList Stream (fetch (TableTopEntry Props) of TableTopEntry))
                    (add NumRecords 1)
                (SETFILEPTR Stream StartLoc)
                (NC.WritePtr Stream NumRecords 3)
                (SETFILEPTR Stream -1)
                3])

```

; Leave file ptr at eof.
 ; Return 3 to indicate this is version 3 substance format.

(NCTableTop.GetFn

[LAMBDA (TableTopCard Length Stream VersionNum)

; Edited 22-Jul-88 00:07 by Trigg

;;; Read in the TableTop card/region pairs. Expect a length field first.

;; rht 5/13/88: Now we're up to version 2 tabletop entry datatype which includes a new Props field.

;; rht 7/22/88: Version 3: Adds notefile uids.

(LET ((NoteFile (NCP.CardNoteFile TableTopCard))
 (NumEntries (NC.ReadPtr Stream 3)))
 (SELECTQ VersionNum

(0 ; Original style table top format.

[for i from 1 to NumEntries
 collect (LET ((UID (NC.ReadUID Stream))
 (Region (NC.ReadRegion Stream))
 Card)
 (if (AND (type? UID UID)
 (NCP.ValidCardP (SETQ Card (NC.CardFromUID UID NoteFile)))
 (REGIONP Region))
 then (create TableTopEntry
 Card _ Card
 Region _ Region
 ShrunkenWinPos _ NIL
 ShrunkenFlg _ NIL]

(1 ; This substance format added shrunken window info.

[for i from 1 to NumEntries
 collect (LET ((UID (NC.ReadUID Stream))
 (Region (NC.ReadRegion Stream))
 (ShrunkenWinPos (CREATEPOSITION (NC.ReadPtr Stream 2)
 (NC.ReadPtr Stream 2)))
 (ShrunkenIndicator (NC.ReadPtr Stream 1))
 Card)
 (if (AND (type? UID UID)
 [NC.CardP (SETQ Card (NC.CardOrCardHolderFromUID UID (fetch (NoteFile
 UID)
 of NoteFile]
 (REGIONP Region)
 (POSITIONP ShrunkenWinPos))
 then (create TableTopEntry
 Card _ Card
 Region _ Region
 ShrunkenWinPos _ ShrunkenWinPos
 ShrunkenFlg _ (EQ ShrunkenIndicator 1)]

(2 ; This substance format adds Props field.

[for i from 1 to NumEntries
 collect (LET ((UID (NC.ReadUID Stream))
 (Region (NC.ReadRegion Stream))
 (ShrunkenWinPos (CREATEPOSITION (NC.ReadPtr Stream 2)
 (NC.ReadPtr Stream 2)))
 (ShrunkenIndicator (NC.ReadPtr Stream 1))
 (Props (NC.ReadPropList Stream))
 Card)
 (if (AND (type? UID UID)
 [NC.CardP (SETQ Card (NC.CardOrCardHolderFromUID UID (fetch (NoteFile
 UID)
 of NoteFile]
 (REGIONP Region)
 (POSITIONP ShrunkenWinPos))
 then (create TableTopEntry
 Card _ Card
 Region _ Region
 ShrunkenWinPos _ ShrunkenWinPos
 ShrunkenFlg _ (EQ ShrunkenIndicator 1)
 Props _ Props]

(3 ; This substance format adds notefile uid.

[for i from 1 to NumEntries
 collect (LET ((UID (NC.ReadUID Stream))
 (NoteFileUID (NC.ReadUID Stream))
 (Region (NC.ReadRegion Stream))
 (ShrunkenWinPos (CREATEPOSITION (NC.ReadPtr Stream 2)
 (NC.ReadPtr Stream 2)))
 (ShrunkenIndicator (NC.ReadPtr Stream 1))
 (Props (NC.ReadPropList Stream))
 Card)
 (if (AND (type? UID UID)
 (NC.CardP (SETQ Card (NC.CardOrCardHolderFromUID UID NoteFileUID)))
 (REGIONP Region)
 (POSITIONP ShrunkenWinPos))
 then (create TableTopEntry
 Card _ Card
 Region _ Region
 ShrunkenWinPos _ ShrunkenWinPos
 ShrunkenFlg _ (EQ ShrunkenIndicator 1)
 Props _ Props]

```
(NCP.ReportError "NCTableTop.GetFn" (CONCAT "Improper substance version num: " VersionNum])
```

)

(DEFINEQ

(NCTableTop.ReplaceUIDsFn

[LAMBDA (TableTopCard HashArray)

; Edited 21-Jul-88 22:53 by Trigg

;; Called under NC.CopyCards, this translates TableTopEntry's Cards according to HashArray.

```
(for TableTopEntry in (NCP.CardSubstance TableTopCard) do (replace (TableTopEntry Card) of TableTopEntry
with (GETHASH (fetch (TableTopEntry Card)
of TableTopEntry)
HashArray])
```

)

(DEFINEQ

(NCTableTop.AddTableTopCard

[LAMBDA NIL

; Edited 29-Mar-89 08:34 by pmi

;; Define the table top card type.

;; rht 1/23/88: No longer allows InsertLink to appear on left button menu.

;; rht 2/4/88: Added

;; pmi 3/29/89: Replaced list of middle button menu items with NCTableTop.MiddleButtonMenuItems

```
(DECLARE (GLOBALVARS NCTableTop.LinkIconBitmap NC.GlobalInsertLinkMenuItem NCTableTop.MiddleButtonMenuItems
))
```

```
(NC.AddCardType 'TableTop 'NoteCard `[(MakeFn ,(FUNCTION NCTableTop.MakeFn))
(EditFn ,(FUNCTION NCTableTop.EditFn))
(GetFn ,(FUNCTION NCTableTop.GetFn))
(PutFn ,(FUNCTION NCTableTop.PutFn))
(QuitFn ,(FUNCTION NIL))
(DirtyPFn ,(FUNCTION NIL))
(MarkDirtyFn ,(FUNCTION NIL))
```

```
`((LinkAnchorModesSupported NIL)
```

```
(DisplayedInMenuFlg T)
```

```
(DefaultHeight 40)
```

```
(DefaultWidth 150)
```

```
(LinkIconAttachedBitMap ,NCTableTop.LinkIconBitmap)
```

```
[LeftButtonMenuItems ,(for Item in (NC.GetCardTypeField LeftButtonMenuItems 'NoteCard)
```

```
collect (if (EQ (CAR Item)
```

```
'Insert% Link)
```

```
then NC.GlobalInsertLinkMenuItem
```

```
else Item]
```

```
(MiddleButtonMenuItems ,NCTableTop.MiddleButtonMenuItems])
```

)

(RPAQQ NCTableTop.LinkIconBitmap



(DECLARE%: EVAL@COMPILE

(RECORD TableTopEntry (Card Region ShrunkenWinPos ShrunkenFlg Props))

(RECORD NCTableTop.BrowserOverviewSpecs (OverviewWinWidth OverviewWinHeight WHERETOATTACHOVERVIEWWIN
OVERVIEWWINMODE))

)

(DECLARE%: DONTEVAL@LOAD

(NCTableTop.AddTableTopCard)

(PUTPROP 'TableTop 'ReplaceUIDsFn (FUNCTION NCTableTop.ReplaceUIDsFn))

(PUTPROP 'TableTop 'ExtraLinkIconMenuItems NCTableTop.MiddleButtonMenuItems)

(PUTPROP 'TableTop 'LinkIconMiddleButtonMenu NIL)

)

(DEFINEQ

(NCAddStub.TableTopCard

[LAMBDA NIL

; Edited 5-Dec-87 20:02 by rht:

(* * Add the tabletop card type stub.)

```
(DECLARE (GLOBALVARS NCTableTop.LinkIconBitmap))
```

```
(NC.AddCardTypeStub 'TableTop 'NoteCard 'NCTABLETOP NIL `((DisplayedInMenuFlg T)
```

```
(LinkIconAttachedBitMap
```

```
,NCTableTop.LinkIconBitmap])
```

)

```
{MEDLEY}<notecards>library>NCTABLETOPCARD.;1
```

```
:: Add a CloseSavingTableTop item to the Close notefile menus.
```

```
(DECLARE%: DONTEVAL@LOAD
```

```
(NCTableTop.FixNoteFileOperationsMenus)  
)
```

FUNCTION INDEX

NCAAddStub.TableTopCard	14	NCTableTop.FixNoteFileOperationsMenus	10
NCTableTop.AddCardsToTableTop	4	NCTableTop.GetFn	13
NCTableTop.AddOpenEventsExpr	9	NCTableTop.MakeFn	11
NCTableTop.AddSelectedCardsToTableTop	9	NCTableTop.MakeOrUpdateNoteFileTableTop	10
NCTableTop.AddTableTopCard	14	NCTableTop.NoteFileWhenOpenedFn	10
NCTableTop.BringUpCardInTableTop	7	NCTableTop.OpenCardInTableTop	2
NCTableTop.BringUpTableTop	2	NCTableTop.ProcessTTEnterProps	8
NCTableTop.CardsInTableTop	4	NCTableTop.PutFn	12
NCTableTop.CloseDownTableTop	3	NCTableTop.RecomputeTableTop	3
NCTableTop.CloseNoteFileSavingTableTop	11	NCTableTop.RemakeTableTop	3
NCTableTop.CloseSessionSavingTableTops	11	NCTableTop.RemakeTableTopSelectingNewCards	8
NCTableTop.CollectCards	4	NCTableTop.RemakeTableTopSelectingNewCardsInOrder	9
NCTableTop.ComputeCardsFromTemplate	6	NCTableTop.RemoveCardsFromTableTop	4
NCTableTop.ComputeTableTop	5	NCTableTop.RemoveSelectedCards	9
NCTableTop.ComputeTableTopUsingTemplate	9	NCTableTop.ReplaceUIDsFn	14
NCTableTop.ComputeTTEnterProps	6	NCTableTop.TableTopBasedP	5
NCTableTop.EditFn	12		

VARIABLE INDEX

NCTableTop.LinkIconBitmap	14	NCTableTop.OldOpenEventsExprs	2
NCTableTop.MiddleButtonMenuItems	2	NCTableTop.OpenEventsExpr	2

RECORD INDEX

NCTableTop.BrowserOverviewSpecs	14	TableTopEntry	14
---------------------------------------	----	---------------------	----

PROPERTY INDEX

NCTABLETOPCARD	2
----------------------	---