NCTableTopCard

By: Randy Trigg (Trigg.pa@Xerox.COM)[1]

Stored: {qv}<notecards>1.3L>Library>NCTableTopCard, .lcom, .ted
Written: July 30, 1988 by Randy Trigg
Last updated: May 5, 1989 by Peggy Irish.
See also NCGuidedTourCard package.

[1] The first version of TableTops was written by Jeff Shrager and subsequently modified by George Cole.

INTRODUCTION

Tabletop cards are a means in NoteCards of capturing the layout of some set of cards on the screen. A tabletop is a snapshot which includes the list of cards, the shapes of their windows, their positions on the screen, the scrolled locations (vertically and possibly horizontally) of the windows' contents, and the order in which to open the windows so that the original (possibly) overlapping arrangement can be preserved. Tabletops have three main uses: (1) as a means of storing card layout across notefile closings/openings. In this case, the system constructs and maintains the tabletop automatically. (2) as aids for online demonstrations of notefiles and to help in creating hard copy screen snapshots for offline presentations. Here, the user creates the tabletop card manually. (3) as "stops" along a NoteCards guided tour. In what follows, I describe the system support for the first two cases. For a description of the use of tabletop cards in guided tours, see the documentation on the NCGuidedTourCard package.

LOADING THE NCTABLETOPCARD PACKAGE

Loading the NCTableTopCard package is done in the same way as loading any other NoteCards card type library package. For example, you can type the following at an Interlisp exec:

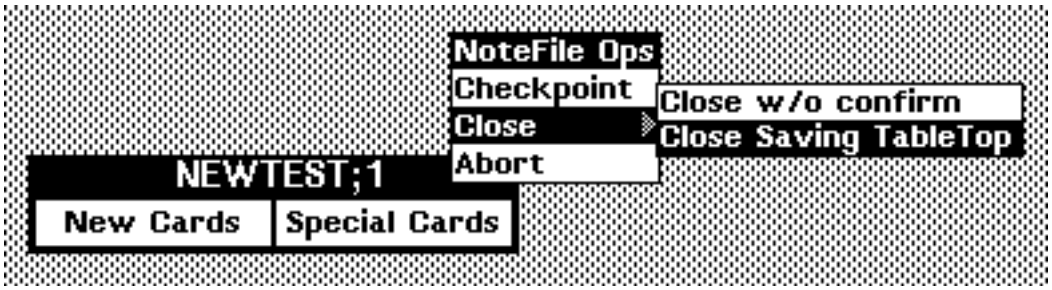
(FILESLOAD (FROM NOTECARDS) NCTABLETOPCARD)

Alternatively, you can add the atom NCTABLETOPCARD to the global var NOTECARDSLIBRARYFILES in your init file. This will ensure that NCTABLETOPCARD is loaded when bringing up a fresh sysout. Finally, if a link to a

tabletop card is followed and the NCTABLETOPCARD package is not currently loaded, then it will be auto-loaded by the system.

SAVING A TABLETOP AT NOTEFILE CLOSING

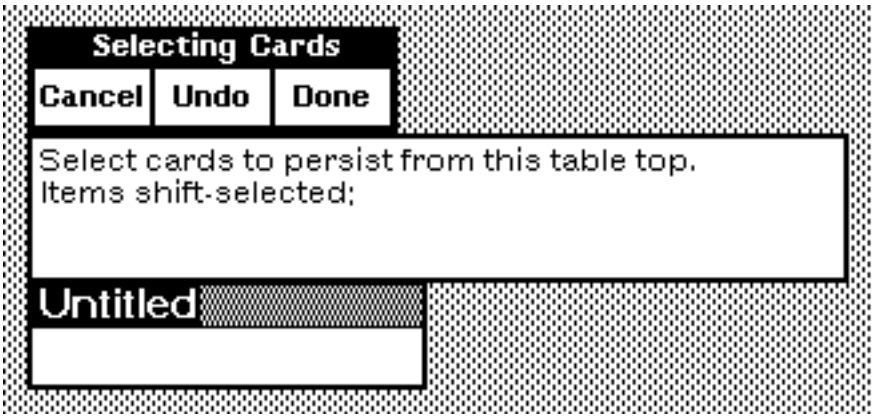
Using a tabletop, one can save the screen layout of cards from a notefile in such a way that the layout can be reinstated the next time the notefile is opened. This allows for greater continuity of work across notefile closings and reopenings. When this package is loaded, the "NoteFile ops" left-button menu available from the notefile icon is augmented slightly. The new pull-across menu for "Close" appears as follows



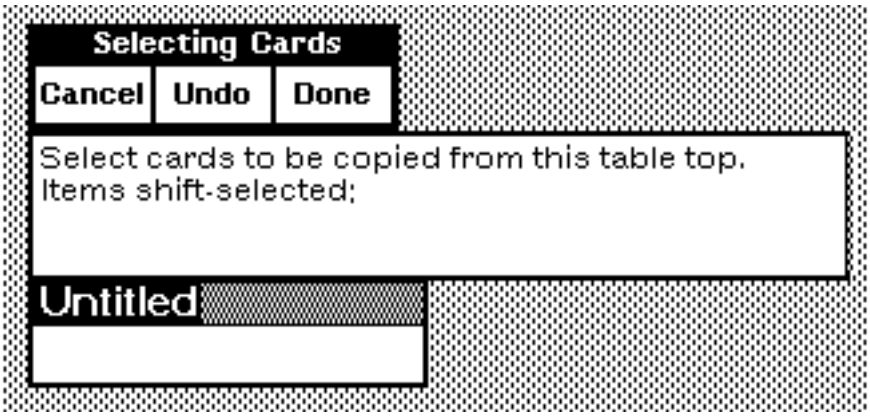
Choosing the option **Close Saving TableTop** causes the current layout of cards with open (or shrunken) windows to be saved in a special tabletop and then the notefile to be closed. The next time the notefile is opened, those same cards will be brought up and arranged (or shrunken) according to the saved layout. (The NCTABLETOPCARD package will be auto-loaded if necessary.)

BUILDING AND MANIPULATING TABLETOPS BY HAND

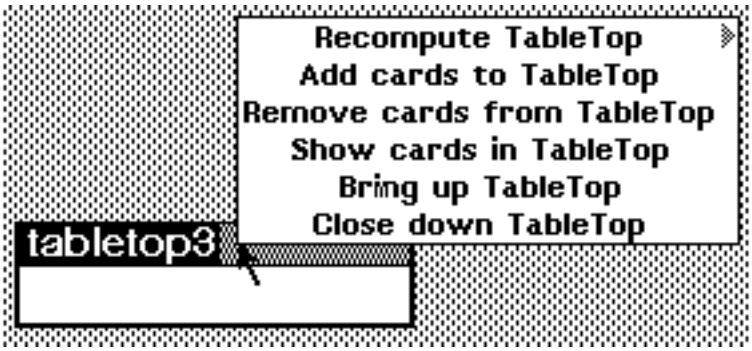
You can create a tabletop card in the usual way, namely by middle-buttoning the NewCards menu on the notefile icon and choosing **TableTop**. You will be asked if you want to use a template tabletop for the creation of this new tabletop. Frequently a user will want several tabletops to share one or more windows, so this option allows one tabletop to be modeled easlily after another. Selecting Yes allows you to pick a tabletop to be used as a template for the new tabletop. Then you will be asked to select the cards to persist from the template, ie. whch cards should actually exist in both tabletops.



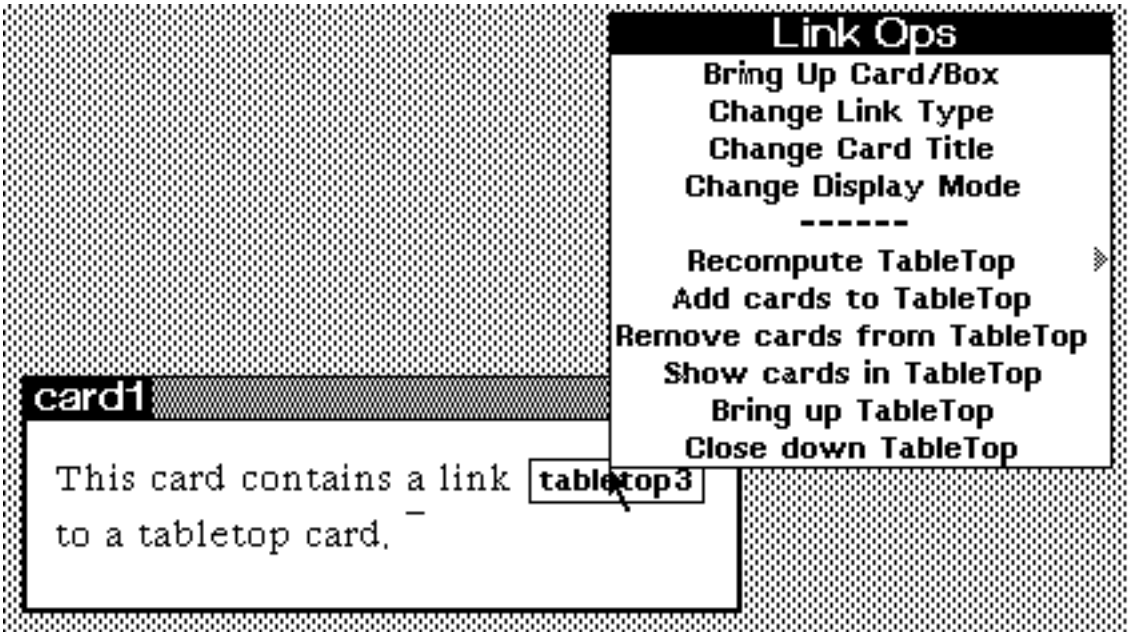
You will also be asked to select the cards to copy from the template. These cards will be copied for the new tabletop, preserving the title, size, shape, location, and default looks of the original cards.



If you don't use a template tabletop, you'll be asked to select a set of cards to form the contents of the tabletop. Select these in the usual way, by holding the shift-key down and left-buttoning in the card windows' titlebars. You'll notice that the tabletop card's window is rather small and has no directly editable substance. Nonetheless, the contents of a tabletop card can be easily modified through a set of tabletop operations available from the card's middle-button titlebar menu.



In addition, each of these operations is available from (the middle-button menu of) any link icon pointing at the tabletop card.



The operations available for tabletop cards are:

Recompute TableTop - Before choosing this operation, you presumably will have rearranged the cards in the tabletop changing some aspect of their layout. Choosing Recompute causes the new layout to be installed in the tabletop card.

Recompute TableTop pull across menu -

New Cards - This operation requires you to specify a new set of cards. That is, the tabletop is recomputed from scratch.

New Cards In Order - This operation requires you to specify a new set of cards, like the **New Cards** menu item. The order in which you choose the new cards is the order in which the cards will appear when you bring up the tabletop.

With Template - This operation asks you to select an existing tabletop to be used as a template for this tabletop.

Add cards to Table Top - This lets you add one or more cards to the tabletop by shift-selecting their windows' titlebars (or links icons pointing at them) in the usual manner. Once the selection is made, the tabletop is recomputed so that the new cards will be ordered correctly relative to the cards already in the tabletop.

Remove cards from Table Top - This brings up a menu containing the titles of all cards in the tabletop. Choose a card to remove by selecting from this menu. A second menu will then pop up containing the remaining cards. In this way, as many cards as desired can be removed from the tabletop. You'll be asked to confirm the removal of all selected cards as soon as you click outside of one of the card title menus.

Show cards in Table Top - This brings up a menu containing the titles of all cards in the tabletop. You can then select from this menu to bring up a single card in the tabletop at its assigned position/shape and scrolled properly.

Bring up Table Top - This opens all cards in the tabletop and positions, shapes, scrolls, shrinks and orders their windows according to the layout information stored in the tabletop. Any cards already open are moved into place and scrolled and shaped (or shrunk) appropriately.

Close down Table Top - This closes any open cards in the tabletop.

The above description covers the standard operation of tabletops. There are, however, two exceptional cases having to do with unopened cards in the tabletop. First, you might try to add a card to the tabletop whose window is not currently open

(or shunken). In this case, the card's most recent position is used and it will appear in the bottom-most position in the window overlap ordering.
Second, you might try to recompute your tabletop even though one or more cards are not currently open. That causes something like the following interactio 

ÓÑ
Of the three choice**s, Can**cel aborts the Recompute operation**. K**eep causes the card to remain in the tabletop. (Note however, that it will be moved to bottom-most in the overlap ordering.**) Rem**ove causes the card to be removed from the tablet**op.**

INFORMATION STORED IN A TABLETOP CARD

As mentioned above, the tabletop card stores layout-related information for each of its cards. This includes: the card window region (i.e. shape and position); the scrolling positions (both vertically and horizontally if any); and whether the card's window is shrunken and if so, the position of its icon. Furthermore, the tabletop keeps the list of cards sorted according to the window overlapping ordering (a.k.a. the TOTOPW ordering). If t**he Recomp**ute sub-co**mmand New Cards In** Order was used to select the cards in the tabletop, then the list of cards is kept in the order specified.
In addition, extra information is kept for certain card types. For browser-based cards, tabletops record information about the shrunken browser overview window (if and where it is attached and its region) and whether the links legend and Editor menu windows are currently open. For Sketch-based cards, tabletops record whether the Sketch editing menu is atta**ched.**

PROGRAMMER'S INTERFACE TO TABLETOP CARDS

To create a new tabletop card programmatically, call NCP.CreateCard as fol**lows:**
(NCP.CreateCard 'TableTop NoteFile Title NoDisplayFlg Props ParentFileBoxes ListOfCards InterestedWindow)
Here, all the parameters are as per the documentation of NCP.CreateCard except that ListOfCards should be a list of (open or shrunken) cards whose layout is to be the tabletop card's initial substance.
The following functions can be used to programmatically manipulate tabletop cards and their cont**ents.**

(NCTableTop.BringUpTableTop TableTopCardOrWindow DoNotUncacheFlg)
The tabletop is brought up, that is, its cards opened and layed out appropriately on the screen. TableTopCardOrWindow can be either a tabletop card object or the window for same. Normally, if the tabletop card itself isn't part of the tabletop being displayed, it is cached, the cards in the tabletop are displayed, and the tabletop card is uncached. A non-NIL DoNotUncacheFlg will prevent the uncaching of the tabletop card (this is useful, for example, when control of caching is desired at the Guided Tour level -- see the NCGuidedTour package documentation).
NCTableTop.BringUpTableTop returns the list of cards it brough**t up.**

(NCTableTop.OpenCardInTableTop TableTopCardOrWindow InterestedWindow)
This causes a single card within a tabletop to be opened and positioned, shaped, etc.

at its proper location. A pop-up menu is used to elicit the card to be opened from the **user.**

(NCTableTop.CloseDownTableTop TableTopCardOrWindow)
Close any open cards in the tabl**etop.**

(NCTableTop.RemakeTableTop TableTopCardOrWindow ListOfCards RetainClosedCardsFlg NoToTopOrder)
Rebuild the tabletop to contain the cards appearing **in ListO**fCards**. If RetainClosedCa**rdsFlg is non-nil, then even closed car**ds in ListO**fCards will be included in the tabletop**. If NoToTo**pOrder is non-NIL, the cards are ordered in the tabletop according to the order in which they appe**ar in ListO**fCards, instead of by their overlapping order on the screen, which is the def**ault.**

(NCTableTop.RecomputeTableTop TableTopCardOrWindow NoToTopOrder)
Recompute the tabletop contents. Normally, the cards in the tabletop are open and arranged in a new arrangement when this call is made**. If NoToTo**pOrder is non-NIL, the order of the cards in the tabletop is not altered according to their overlapping order on the screen, which is the def**ault.**

(NCTableTop.RemoveCardsFromTableTop TableTopCardOrWindow CardsToRemove)
Remove cards in the **list Card**sToRemove from the tabletop's substance. Returns the cards actually **removed.**

(NCTableTop.AddCardsToTableTop TableTopCardOrWindow CardsToAdd InterestedWindow)
Add cards appe**aring in C**ardsToAdd to the substance of the t**abletop.**

(NCTableTop.CardsInTableTop Table TopCard)
Return the list of cards currently appearing in the t**abletop.**

(NCTableTop.TableTopBasedP CardOrCardType)
Return no**n-nil if CardO**rCardType is either a tabletop card or a card of a type that inherits from tabletop card. (It does the obvious **thing if CardO**rCardType is a car**d type.)**

(NCTableTop.CollectCards RootTableTopCards LinkTypes MaxDepth FollowCrossFileLinksFlg)
This function is analogous to NCP.CollectCards. It starts from a list of tabletop cards **given by RootTabl**eTopCards and computes the transitive closure by following links of types appe**aring on** LinkTypes to a maximum depth spec**ified by** MaxDepth where the depth 1 cards are the cards directly in the tabletop su