

File created: 17-Aug-87 12:55:11 {QV}<NOTECARDS>1.3L>LIBRARY>NCSEEDITCARD.;20

changes to: (VARS NCSEEDITCARDCOMS)

previous date: 16-Jun-87 18:21:36 {QV}<NOTECARDS>1.3L>LIBRARY>NCSEEDITCARD.;19

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::
:: Copyright (c) 1986, 1987 by Xerox Corporation. All rights reserved.

(RPAQQ **NCSEEDITCARDCOMS**

```
(( * File created by MILLER)
(P (if (EQ MAKESYSNAME 'KOTO)
      then
        (FILESLOAD (FROM {PHYLUM}<sedit>koto>)
                    SEDIT)))
;; For Lyric, we'll need a (DECLARE: EVAL@COMPILE (LOADCOMP 'SEEDIT-DECLS))
(FILESD (FROM VALUEOF NOTECARDSDIRECTORIES)
        SEDIT-IMAGEOBJ)
(FNS NC.AddSEditCard NC.GetSEditCard NC.MakeSEditCard NC.PutSEditCard NC.EditSEditCard NC.QuitSEditCard
      NC.MarkDirtySEditCard NC.SEditCardDirtyP NC.SEditCardToTextStream NC.SEditUpdateSubstance)
(BITMAPS NC.SEditLinkIcon)
(P (NC.AddSEditCard))
(PROP ExportSubstanceFn SEdit)
;; the following belong in programmers interface to SEdit
(FNS EditContextWindow EditContextOpenNode EditContextProfile ChangedStructure?)
```

;;; no longer needed?

```
(FNS NC.SEditMETAxAdvice)
;; (p (advise '\\complete|'after '(NC.SEditMETAxAdvice)))
))
```

(* * File created by MILLER)

```
(if (EQ MAKESYSNAME 'KOTO)
    then (FILESLOAD (FROM {PHYLUM}<sedit>koto>)
              SEDIT))
```

:: For Lyric, we'll need a (DECLARE: EVAL@COMPILE (LOADCOMP 'SEEDIT-DECLS))

```
(FILESLOAD (FROM VALUEOF NOTECARDSDIRECTORIES)
           SEDIT-IMAGEOBJ)
```

(DEFINEQ

(NC.AddSEditCard

[LAMBDA NIL

(* MarkM " 6-Nov-86 19:57")

;; SEdit Card is for editing Lisp Lists. Is not a specialization of the List Card, because the only thing we're interested in inheriting is the CopyFn
;; which should be generic anyway

```
(NC.AddCardType 'SEdit 'NoteCard `((MakeFn %, (FUNCTION NC.MakeSEditCard))
                                   (EditFn %, (FUNCTION NC.EditSEditCard))
                                   (QuitFn %, (FUNCTION NC.QuitSEditCard))
                                   (GetFn %, (FUNCTION NC.GetSEditCard))
                                   (PutFn %, (FUNCTION NC.PutSEditCard))
                                   (CopyFn %, (FUNCTION NC.ListCardCopyFn))
                                   (MarkDirtyFn %, (FUNCTION NC.MarkDirtySEditCard))
                                   (DirtyPFn %, (FUNCTION NC.SEditCardDirtyP)))
  `((LinkDisplayMode Title)
    (DefaultHeight 400)
    (DefaultWidth 350)
    (LinkAnchorModesSupported Global)
    (DisplayedInMenuFlg T)
    (LinkIconAttachedBitMap ,NC.SEditLinkIcon])
```

(NC.GetSEditCard

[LAMBDA (Card Length Stream)

(* MarkM " 6-Nov-86 19:57")

;; Get the list substance from the disk. We have to use HREAD instead of READ because of the possibly contained Image-Objects

```
(HREAD Stream))
```

(NC.MakeSEditCard

[LAMBDA (Card Title NoDisplayFlg Substance)

; Edited 16-Apr-87 12:15 by kirk:

```
;; Make an SEdit Card. If no substance is provided, start with a dummy:
[OR Substance (SETQ Substance (COPY '("empty"))
(NC.SetSubstance Card Substance)
(if NoDisplayFlg
  then Card
  else (NCP.ApplyCardTypeFn EditFn Card Substance NIL])
```

(NC.PutSEditCard

; Edited 16-Jun-87 18:18 by kirk:

```
[LAMBDA (Card Stream)
;; kirk 3Dec86: added \close.open.node call.
;;
;; first make sure the edit completed and save the profile
[LET [(Context (WINDOWPROP (NC.FetchWindow Card)
'EditContext)
(if Context
  then (if (EditContextOpenNode Context)
    then (\close.node Context))
    ;; Now lets save the profile (package, readable, etc.)
    (NCP.CardProp Card 'PROFILE (EditContextProfile Context)
;; Put list substance. Must use HPRINT instead of PRINT because the list may contain Image-Objects.
(HPRINT (NC.FetchSubstance Card)
Stream T)
0])
```

(NC.EditSEditCard

; Edited 16-Jun-87 15:14 by kirk:

```
[LAMBDA (Card Substance Region/Position)
(LET ((Window (NC.FetchWindow Card)))
[if Window
  then (AND (WINDOWP (WINDOWPROP Window 'ICONFOR))
    (SETQ Window (EXPANDW Window)))
    (TOTOPW Window)
    (FLASHW Window)
  else (LET* [Context (Region (NC.DetermineDisplayRegion Card Region/Position))
    (Title (CONCAT (NCP.CardTitle Card)))
    (Profile (NCP.CardProp Card 'PROFILE))
    (SEditPropList (APPEND (LIST 'NAME Title)
      (if Profile
        then (LIST 'PROFILE Profile)
        (if (EQ MAKESYSNAME 'KOTO)
          then (push \SEdit.Regions Region)
          (SETQ Context (SEdit Substance (LIST 'NAME Title)))
          else (push \regions Region)
          (SETQ Context (SEdit Substance SEditPropList :DONTWAIT)))
      (replace (EditContext RootChangedFn) of Context with (LIST 'NC.SEditUpdateSubstance Card))
    ;; set the RootChangedFn in the sedit context to update the substance pointer in the card's cache .
    (SETQ Window (EditContextWindow Context)
Window])
```

(NC.QuitSEditCard

; Edited 16-Jun-87 18:17 by kirk:

```
[LAMBDA (CardIdentifier)
;; SEdit can be shut down by closing its window
(LET* ((Card (NC.CoerceToCard CardIdentifier))
(Window (NC.FetchWindow Card))
(CLOSEW Window)
(NC.DeactivateCard Card))
```

(NC.MarkDirtySEditCard

(* kirk%: "16-Dec-86 14:08")

```
[LAMBDA (Card ResetFlg)
;; Mark or unmark List as having been changed
;; Kirk 16Dec86: changed to use property hung off of window instead of SEdit ChangedStructure? flag.
(WINDOWPROP (NCP.CardWindow Card)
'Changed
(NOT ResetFlg])
```

(NC.SEditCardDirtyP

; Edited 25-Jan-87 21:06 by kirk:

```
[LAMBDA (Card)
;; Has the List structure been changed?
(LET* [(Window (NC.FetchWindow Card))
(Context (WINDOWPROP Window 'EditContext)
(OR (AND Context (ChangedStructure? Context))
(WINDOWPROP (NCP.CardWindow Card)
'Changed])
```

(NC.SEditCardToTextStream

[LAMBDA (substance)

(* MarkM " 6-Nov-86 20:02")

:: I am the ExportSubstanceFn method for SEdit cards. When creating a Document card, I will return a textStream on which the list is
:: pretty-printed. It would be nice if I knew the ParaLooks at the current point in the document, so I could adapt to its width. (All the logic for this is
:: already in \pretty.print.to.tedit, you just have to pass the enclosing tedit as the second argument to it)

((\pretty.print.to.tedit substance))

(NC.SEditUpdateSubstance

[LAMBDA (NEWSTRUCTURE CARD)

; Edited 9-Feb-87 10:41 by kirk:

(NC.SetSubstance CARD NEWSTRUCTURE))

)

(RPAQQ **NC.SEditLinkIcon**



(NC.AddSEditCard)

(PUTPROPS **SEdit ExportSubstanceFn** NC.SEditCardToTextStream)

:: the following belong in programmers interface to SEdit

(DEFINEQ

(EditContextWindow

[LAMBDA (Context)

; Edited 18-Jan-87 23:26 by kirk:

(**fetch** DisplayWindow **of** Context))

(EditContextOpenNode

[LAMBDA (CONTEXT)

; Edited 9-Feb-87 11:58 by kirk:

(**fetch** OpenNode **of** CONTEXT))

(EditContextProfile

[LAMBDA (Context)

; Edited 16-Jun-87 11:12 by kirk:

(**fetch** (EditContext Profile) **of** Context))

(ChangedStructure?

[LAMBDA (Context)

; Edited 23-Jan-87 02:06 by kirk:

(**fetch** ChangedStructure? **of** Context))

)

::: no longer needed?

(DEFINEQ

(NC.SEditMETAxAdvice

[LAMBDA NIL

; Edited 23-Jan-87 02:27 by kirk:

(* * makes sure Close Card still saves your work right after a METAx and adjusts the region to fit the extent.)

(LET ((Window (EditContextWindow context))
Card Extent Region)
if (SETQ Card (NCP.CardFromWindow Window))
then (NC.MarkDirtySEditCard Card))

)

:: (p (advise '|\\complete| 'after '(|NC.SEditMETAxAdvice|)))

(PUTPROPS **NCSEDCARD COPYRIGHT** ("Xerox Corporation" 1986 1987))

FUNCTION INDEX

ChangedStructure?3	NC.AddSEditCard1	NC.MarkDirtySEditCard ...2	NC.SEditCardToTextStream 3
EditContextOpenNode3	NC.EditSEditCard2	NC.PutSEditCard2	NC.SEditMETaxAdvice3
EditContextProfile3	NC.GetSEditCard1	NC.QuitSEditCard2	NC.SEditUpdateSubstance .3
EditContextWindow3	NC.MakeSEditCard1	NC.SEditCardDirtyP2	

VARIABLE INDEX

NC.SEditLinkIcon3	NCSEDITCARDOMS1
-------------------------	-----------------------

PROPERTY INDEX

SEdit3
