

File created: 20-Mar-2024 17:07:35 {DSK}<home>frank<il>notecards>system>NOTECARDS.;14

edit by: frank

changes to: (FNS NC.SetUpNOTECARDSDIRECTORIES)
(VARS NOTECARDSCOMS)

previous date: 20-Mar-2024 17:02:38 {DSK}<home>frank<il>notecards>system>NOTECARDS.;13

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

(RPAQQ **NOTECARDSCOMS**

```
((GLOBALVARS *NC.ReleaseIdentifier* NOTECARDSDIRECTORIES \NC.SourceAccessFlg *NC.LoadedFiles*  
  UseMapWithPoliticalBoundaries AFTERLOGOUTFORMS)  
(INITVARS (*NC.LoadedFiles* NIL)  
  (*NC-LOAD-RELEASE* NIL))  
(VARS (*NC.ReleaseIdentifier* "2.0"))
```

;;; Set up the NoteCards directories globalvars

```
(FILES (SYSLOAD)  
  PSEUDOHOSTS)  
(FNS NC.SetUpNOTECARDSDIRECTORIES)  
(DECLARE%: DONTEVAL@LOAD DOCOPY (P (NC.SetUpNOTECARDSDIRECTORIES)))  
[P (IF (NOT (CL:FIND-PACKAGE "ROOMS"))  
  THEN  
    (DEFPACKAGE "ROOMS" (:EXPORT "INTERACTIVE-CLOSE-WINDOW" "ALL-WINDOWS" "*CURRENT-ROOM*"  
      "*ROOMS-SYSTEM-DATE*" "HIDE-WINDOW" "UN-HIDE-WINDOW" "WINDOW-HIDDEN?")]
```

;;; Load a few LIBRARYFILES

```
(DECLARE%: DONTEVAL@LOAD DOCOPY (FILES (SYSLOAD)  
  FILEBROWSER TEDIT HRULE GRAPHER))
```

;;; card types and library packages

```
(GLOBALVARS NOTECARDTYPESTUBS NOTECARDTYPES *NC.LibraryFiles* *NC.LispUsersFiles*)  
(ADDVARS (NOTECARDTYPESTUBS Sketch Graph Browser Search Document LinkIndex))  
(ADDVARS (NOTECARDTYPES Hash List Text FileBox))  
(ADDVARS (*NC.LibraryFiles* NCHACKS NEWSHADEITEM))  
(ADDVARS (*NC.LispUsersFiles* SCALEBITMAP STYLESHEET))
```

;;; Load the Library Files

```
(DECLARE%: DONTEVAL@LOAD DOCOPY (FILES (SYSLOAD FROM VALUEOF NOTECARDSDIRECTORIES)  
  NCHACKS NEWSHADEITEM TEDIT-PROCESS-KILLER)  
(FILES (SYSLOAD FROM VALUEOF LISPUSERSDIRECTORIES)  
  STYLESHEET))
```

;;; code module lists

```
(GLOBALVARS NC.CoreFiles NC.Files)
```

;;; global params

```
(GLOBALVARS NC.DeletedLinkImageObject NC.DeletedLinkImageFns NC.InvisibleLinkImageObject  
  NC.InvisibleLinkImageFns NC.LinkIconImageFns NC.NoDeleteImageFns NC.SystemDate  
  TEDIT.INPUT.FORMATS WindowMenuCommands NCINITIALGLOBALPARAMS)  
(INITVARS NCINITIALGLOBALPARAMS)
```

;;; Load the NoteCards files

```
(FNS NC.FindFile NC.LoadFileFromDirectories NC.LookupAutoloadFnFile NC.StoreAutoloadFnFile  
  SIMPLE-FILENAME SIMPLE-LOADEDFILELST)  
[DECLARE%: DONTEVAL@LOAD DOCOPY (FILES (SYSLOAD FROM VALUEOF NOTECARDSDIRECTORIES)  
  NCCONFIG)  
  ; (P (ADVISE 'WHEREIS 'BEFORE NIL '(if (GETPROP  
  ; 'NOTECARDS 'FILEDATES) then (NC.FixFileDates  
  ; 'NOTECARDS) (UNADVISE 'WHEREIS))))  
(P (for File in NC.CoreFiles do (NC.LoadFileFromDirectories File]
```

;;; Initialization functions

```
(FNS NC.PostGreet NC.LoadPatches)  
(ADDVARS (POSTGREETFORMS (NC.PostGreet)))  
(DECLARE%: DONTEVAL@LOAD DOCOPY (P (NC.PostGreet))
```

```
{MEDLEY}<notecards>system>NOTECARDS.;1 (NOTECARDSCOMS cont.)
(NC.LoadPatches))
```

Page 2

;;; Set up Lafite Bug Report Forms

```
(FNS NC.MakeLafiteForm)
(ADDVARS (LAFITESPECIALFORMS ("NoteCards Report" (FUNCTION NC.MakeLafiteForm)
"Report bug or request new feature for NoteCards.")))
(VARS (LAFITEFORMSMENU NIL))
; force menu to be recomputed
```

;;; INITIALIZE NoteCards

```
(FNS NC.Setup)
(DECLARE%: DONTEVAL@LOAD DOCOPY (P (NC.Setup)))
(ADDVARS (BackgroundMenuCommands (NoteCards (NC.BringUpNoteCardsIcon)
"Bring up the NoteCards control icon.")))
(VARS (BackgroundMenu NIL))
; force menu to be recomputed
(FNS NC.SetSystemDate)
(DECLARE%: DONTEVAL@LOAD DOCOPY (P (NC.SetSystemDate)))
(PROP (FILETYPE MAKEFILE-ENVIRONMENT)
NOTECARDS))
```

```
(DECLARE%: DOEVAL@COMPILE DONTCOPY
```

```
(GLOBALVARS *NC.ReleaseIdentifier* NOTECARDSDIRECTORIES \NC.SourceAccessFlg *NC.LoadedFiles*
UseMapWithPoliticalBoundaries AFTERLOGOUTFORMS)
)
```

```
(RPAQ? *NC.LoadedFiles* NIL)
```

```
(RPAQ? *NC-LOAD-RELEASE* NIL)
```

```
(RPAQ *NC.Releaseldentifier* "2.0")
```

;;; Set up the NoteCards directories globalvars

```
(FILESLOAD (SYSLOAD)
PSEUDOHOSTS)
```

```
(DEFINEQ
```

(NC.SetUpNOTECARDSDIRECTORIES

```
[LAMBDA (BaseDirectory ReleaseIdentifier)
(DECLARE (GLOBALVARS NOTECARDSDIRECTORIES \NC.SourceAccessFlg *NC.ReleaseIdentifier*))
; Edited 20-Mar-2024 17:07 by frank
; Edited 19-May-2021 17:04 by pi
```

;; Starting from the directory this file is being loaded from, set up NOTECARDSDIRECTORIES with the right list of directories for loading
;; NoteCards files from. NB that this function must be run from a loading file to work properly, as it uses (INPUT) to decide what directory to use.

```
(SETQ NOTECARDSDIRECTORIES (LET [NoteCardsBaseDirectory StrPtr NoteCardsLoadDirectory
(NoteCardsSubDirectories '(system library lispusers patches]
```

;; Get the directory for the file now being loaded

```
[COND
(BaseDirectory (SETQ NoteCardsBaseDirectory (TRUEFILENAME BaseDirectory))
)
(T [SETQ NoteCardsLoadDirectory (TRUEFILENAME (PACKFILENAME.STRING
'HOST
(FILENAMEFIELD
(FULLNAME (INPUT))
'HOST)
'DIRECTORY
(FILENAMEFIELD
(FULLNAME (INPUT))
'DIRECTORY]
```

;; Find the directory path up to the first subdirectory called NoteCards

```
(COND
((SETQ StrPtr (STRPOS "NOTECARDS" (U-CASE NoteCardsLoadDirectory)
NIL NIL NIL T NIL T))
(SETQ NoteCardsBaseDirectory (SUBATOM NoteCardsLoadDirectory 1
StrPtr]
```

;; Create the directory search list based on the NoteCardsBaseDirectory

```
(COND
[NoteCardsBaseDirectory
(APPEND (if NoteCardsLoadDirectory
then (LIST NoteCardsLoadDirectory)
else (LIST))
(MAPCAR NoteCardsSubDirectories
#' (LAMBDA (SUBDIR)
(PACKFILENAME.STRING 'HOST (FILENAMEFIELD
NoteCardsBaseDirectory
```

```

'HOST)
'DIRECTORY
(CONCAT (FILENAMEFIELD NoteCardsBaseDirectory
'DIRECTORY)
">" SUBDIR]
(T (LIST NoteCardsLoadDirectory])

)

(DECLARE%: DONTEVAL@LOAD DOCOPY

(NC.SetUpNOTECARDSDIRECTORIES)
)

(IF (NOT (CL:FIND-PACKAGE "ROOMS"))
THEN (DEFPACKAGE "ROOMS" (:EXPORT "INTERACTIVE-CLOSE-WINDOW" "ALL-WINDOWS" "**CURRENT-ROOM*"
"*ROOMS-SYSTEM-DATE*" "HIDE-WINDOW" "UN-HIDE-WINDOW" "WINDOW-HIDDEN?"))))

;;; Load a few LIBRARYFILES

(DECLARE%: DONTEVAL@LOAD DOCOPY

(FILESLoad (SYSLOAD)
FILEBROWSER TEDIT HRULE GRAPHER)
)

;;; card types and library packages

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS NOTECARDTYPESTUBS NOTECARDTYPES *NC.LibraryFiles* *NC.LispUsersFiles*)
)

(ADDTovar NOTECARDTYPESTUBS Sketch Graph Browser Search Document LinkIndex)
(ADDTovar NOTECARDTYPES Hash List Text FileBox)
(ADDTovar *NC.LibraryFiles* NCHACKS NEWSHADEITEM)
(ADDTovar *NC.LispUsersFiles* SCALEBITMAP STYLESHEET)

;;; Load the Library Files

(DECLARE%: DONTEVAL@LOAD DOCOPY

(FILESLoad (SYSLOAD FROM VALUEOF NOTECARDSDIRECTORIES)
NCHACKS NEWSHADEITEM TEDIT-PROCESS-KILLER)

(FILESLoad (SYSLOAD FROM VALUEOF LISPUSERSDIRECTORIES)
STYLESHEET)
)

;;; code module lists

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS NC.CoreFiles NC.Files)
)

;;; global params

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS NC.DeletedLinkImageObject NC.DeletedLinkImageFns NC.InvisibleLinkImageObject
NC.InvisibleLinkImageFns NC.LinkIconImageFns NC.NoDeleteImageFns NC.SystemDate TEDIT.INPUT.FORMATS
WindowMenuCommands NCINITIALGLOBALPARAMS)
)

(RPAQ? NCINITIALGLOBALPARAMS NIL)

;;; Load the NoteCards files

(DEFINEQ

(NC.FindFile
[LAMBDA (FileName TypeName CheckConnectedDirFlg SourceFlg Directories)
; Edited 19-May-2021 17:00 by pi
(DECLARE (GLOBALVARS NOTECARDSDIRECTORIES))
(LET [(FilesLoadDirs (if Directories
then (EVAL Directories)
elseif CheckConnectedDirFlg

```

```

      then (CONS (DIRECTORYNAME T T)
                 NOTECARDSDIRECTORIES)
    else NOTECARDSDIRECTORIES))
  (FileProps (AND *NC-LOAD-RELEASE* (GETPROP FileName (if SourceFlg
                                                           then 'SourceFile
                                                           else 'CompiledFile]
[OR FileName (SETQ FileName (PACK* 'NC (U-CASE TypeName)
                                   'CARD]
(OR (AND (NOT NC.InstallFlg)
         FileProps
         (FINDFILE FileProps))
  (if (AND (NULL (FILENAMEFIELD FileName 'EXTENSION))
        (FINDFILE (PACKFILENAME 'BODY FileName 'EXTENSION (if SourceFlg
                                                                then ""
                                                                else COMPILE.EXT))
        T FilesLoadDirs))
    else (FINDFILE FileName T FilesLoadDirs])

```

(NC.LoadFileFromDirectories

[LAMBDA (FileName Directories QuietFlg TypeName AlwaysFlg) ; Edited 2-Oct-90 15:53 by gadener

```

;; Try to load FileName from Directories. The latter can be a list in which case the elements should be directory paths. It can be an atom in which
;; case it should be a DIRECTORIES var. Or it can be NIL in which case it defaults to the litatom NOTECARDSDIRECTORIES.
;; rg 9/4/87 now pushes file name onto NC.LoadedFiles for use by NC.PostGreet
;; rg 1/15/88 now checks rootified version of LOADEDFILELST, won't load any file that it finds there. Net effect is that this fn will NEVER load a file
;; that has been loaded in ANY version previously. No longer calls NC.FixFileDates.
;; rg 1/22/88: doesn't SYSLOAD if NC.SourceAccessFlg set, returns full file name instead of NC.LoadedFiles. Trimmed change log.
;; rg 5/2/88: Uses FILEDATES prop instead of checking SIMPLELOADEDFILELST, added AlwaysFlg
;; rg 6/22/88 put search through LOADEDFILELST back in - FILEDATES property check wasn't sufficient for, e.g., stubs.
;; pmi 9/8/88: Changed printout "Attempting to autoloading ..." so that it prints either the FileName or the TypeName, in case FileName is NIL. Also
;; changed to add the name part of FullName instead of FileName to the NC.LoadedFiles list.

```

```

(DECLARE (GLOBALVARS *NC.LoadedFiles* \NC.SourceAccessFlg))
(RESETLST
  [RESETSAVE NIL (if (OPENWP (WFROMDS (TTYDISPLAYSTREAM)
                                   T))
                    then (SETQ QuietFlg T)
                    ' (NIL))
    else '(CLOSEW , (WFROMDS (TTYDISPLAYSTREAM)
                              T))
  [RESETSAVE NIL (if QuietFlg
                    then ' (NIL))
    else '(WINDOWPROP , (WFROMDS (TTYDISPLAYSTREAM)
                                  'PAGEFULLFN
                                  , (WINDOWPROP (WFROMDS (TTYDISPLAYSTREAM)
                                                          'PAGEFULLFN
                                                          'NIL))
                                  'NIL))
  [LET ((Window (WFROMDS (TTYDISPLAYSTREAM)))
        (SimpleName (SIMPLE-FILENAME FileName))
        (FullName))
    (COND
      ((AND [OR AlwaysFlg (NOT (GETPROP SimpleName 'FILEDATES))
              (for LoadedFile in LOADEDFILELST never (EQ SimpleName (SIMPLE-FILENAME LoadedFile]
              (SETQ FullName (NC.FindFile FileName TypeName TypeName NIL (OR (GETPROP FileName
                                          'Directories)
                                          Directories]
              (if (NOT QuietFlg)
                  then (CLEARW Window)
                      (FLASHWINDOW Window)
                      (if TypeName
                          then (PRINTOUT Window "Card type " TypeName " not loaded." T))
                          (PRINTOUT Window "Attempting to autoloading " (OR FileName TypeName)
                          T))
              (PROG1 [if AlwaysFlg
                      then (LOAD FullName (if (NOT \NC.SourceAccessFlg)
                                              then 'SYSLOAD))
                      else (APPLY* #'FILESLOAD (if \NC.SourceAccessFlg
                                                  then '(,FullName)
                                                  else '( (SYSLOAD)
                                                  ,FullName)
                      (pushnew *NC.LoadedFiles* (FILENAMEFIELD FullName 'NAME))))))

```

(NC.LookupAutoloadFnFile

[LAMBDA (AutoloadFn) ; Edited 14-Jan-88 11:53 by Randy.Gobbel

```

;; rg 1/14/88 no longer gets fully-qualified name, lets NC.LoadFileFromDirectories load from connected dir if NC.InstallFlg set
;; Fetch the filename for loading given function
(GETPROP AutoloadFn 'File])

```

(NC.StoreAutoloadFnFile

[LAMBDA (AutoloadFn FileName Directories) (* edited%: "26-Aug-87 17:22")

(* * Store an entry in the autoload table for given AutoloadFn. It consists of FileName and Directories variable or list or NIL.)

```
(PUTPROP AutoloadFn 'File FileName)
(PUTPROP FileName 'Directories Directories)
(pushnew NC.AutoloadableFns AutoloadFn))
```

(SIMPLE-FILENAME

```
[LAMBDA (file)
  (LET [(ext (FILENAMEFIELD file 'EXTENSION))
        (PACKFILENAME 'NAME (FILENAMEFIELD file 'NAME)
                        'EXTENSION)
        (if (NOT (MEMB ext *COMPILED-EXTENSIONS*))
            then ext)])
```

; Edited 20-Jan-88 11:22 by Randy.Gobbel

(SIMPLE-LOADEDFILEST

```
[LAMBDA NIL
  (for file in LOADEDFILEST collect (SIMPLE-FILENAME file))
```

; Edited 20-Jan-88 11:23 by Randy.Gobbel

)

(DECLARE%: DONTVAL@LOAD DOCOPY

```
(FILESLOAD (SYSLOAD FROM VALUEOF NOTECARDSDIRECTORIES)
  NCCONFIG)
```

```
(for File in NC.CoreFiles do (NC.LoadFileFromDirectories File))
)
```

;;; Initialization functions

(DEFINEQ

(NC.PostGreet

```
[LAMBDA NIL
  ;; called when loading NOTECARDS and after each greet (for NOTECARDS sysout users) Note this means you cannot put anything in here that
  ;; should not be called twice!
  (DECLARE (GLOBALVARS \NC.SourceAccessFlg NC.CardTypes *NC.LoadedFiles* NOTECARDTYPES NOTECARDTYPESTUBS
                    *NC.LibraryFiles* *NC.LispUsersFiles* NCINITIALGLOBALPARAMS))
  [LET (FullFileName TypeName Type File)
    (for TypeName in NOTECARDTYPES unless (AND (SETQ Type (GETHASH TypeName NC.CardTypes))
                                                (NOT (fetch (NoteCardType StubFlg) of Type)))
      do (NC.CardTypeLoader TypeName NIL T))
    ;; in case new patches have been made since the sysout was made or the last greet
    (for TypeName in NOTECARDTYPESTUBS unless (OR (NC.CardTypeRecord TypeName)
                                                  (MEMB TypeName NOTECARDTYPES)
                                                  (GETHASH TypeName NC.CardTypes))
      do (NC.CardTypeStubLoader TypeName))
    ;; Load the library packages requested by the user's NOTECARDSLIBRARYFILES parameters
    (for File in *NC.LibraryFiles* unless (GETPROP File 'FILEDATES)
      do (COND
          ((SETQ FullFileName (NC.FindFile File))
           (NC.LoadFileFromDirectories FullFileName))
          (T (PRINT (CONCAT "NoteCards: Can't find library file: " File ".")
                    (NC.LoadFileFromDirectories File 'LISPUSERSDIRECTORIES)))
          (if (GETD 'START-TEDIT-KILLER)
              then (START-TEDIT-KILLER))
          (if (AND \NC.SourceAccessFlg (GETD 'XCL::ADD-WHERE-IS-DATABASE))
              then (XCL::ADD-WHERE-IS-DATABASE (NC.FindFile 'NOTECARDS.HASH NIL NIL T)))
          (NCP.NoteCardsParameters NCINITIALGLOBALPARAMS]))
```

; Edited 10-Jun-2021 14:37 by pi

(NC.LoadPatches

```
[LAMBDA NIL
  (DECLARE (GLOBALVARS NOTECARDSDIRECTORIES))
  ;; Load the POSTRELEASEPATCHES File, which will get the latest patches.
  (if (NC.FindFile 'POSTRELEASEPATCHES)
      then (NC.LoadFileFromDirectories 'POSTRELEASEPATCHES NIL NIL NIL T))
```

; Edited 19-May-2021 17:12 by pi

)

(ADDTovar POSTGREETFORMS (NC.PostGreet))

(DECLARE%: DONTVAL@LOAD DOCOPY

(NC.PostGreet)

(NC.LoadPatches)

)

;;; Set up Lafite Bug Report Forms

[illegible]

```
{MEDLEY}<notecards>system>NOTECARDS.;1 (NC.Setup cont.)
```

Page 7

```
(FUNCTION NIL)))  
(SETQ NC.InvisibleLinkImageObject (IMAGEOBJCREATE "InvisibleLink" NC.InvisibleLinkImageFns])  
)
```

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

```
(NC.Setup)  
)
```

```
(ADDTovar BackgroundMenuCommands (NoteCards (NC.BringUpNoteCardsIcon)  
"Bring up the NoteCards control icon."))
```

```
(RPAQQ BackgroundMenu NIL)
```

```
:: force menu to be recomputed
```

```
(DEFINEQ
```

```
(NC.SetSystemDate
```

```
[LAMBDA NIL
```

```
(* edited%: "26-Aug-87 18:30")
```

```
(* * Set the NC.SystemDate from the stored file dates)
```

```
(DECLARE (GLOBALVARS NC.SystemDate NC.Files))
```

```
(LET (ThisFileDate)
```

```
(SETQ NC.SystemDate "23-Aug-70 13:28:52")
```

```
(for File in (LDIFFERENCE NC.Files ' (NCCONFIG POSTRELEASEPATCHES))
```

```
do [SETQ ThisFileDate (OR (CAAR (GETPROP File 'FILEDATES))
```

```
(GETPROP File 'FileDate]
```

```
(if (AND ThisFileDate (GREATERP (IDATE ThisFileDate)
```

```
(IDATE NC.SystemDate)))
```

```
then (SETQ NC.SystemDate ThisFileDate)
```

```
(PUTPROP 'NC.SystemDate 'NewestFile File])
```

```
)
```

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

```
(NC.SetSystemDate)
```

```
)
```

```
(PUTPROPS NOTECARDS FILETYPE :FAKE-COMPILE-FILE)
```

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

FUNCTION INDEX

NC.FindFile	3	NC.MakeLafiteForm	6	NC.SetupNOTECARDSDIRECTORIES	2
NC.LoadFileFromDirectories	4	NC.PostGreet	5	NC.StoreAutoloadFnFile	4
NC.LoadPatches	5	NC.SetSystemDate	7	SIMPLE-FILENAME	5
NC.LookupAutoloadFnFile	4	NC.Setup	6	SIMPLE-LOADEDFILELST	5

VARIABLE INDEX

NC-LOAD-RELEASE	2	*NC.ReleaseIdentifier* ..	2	LAFITESPECIALFORMS	6	POSTGREETFORMS	5
NC.LibraryFiles	3	BackgroundMenu	7	NCINITIALGLOBALPARAMS ..	3		
NC.LispUsersFiles	3	BackgroundMenuCommands ..	7	NOTECARDTYPES	3		
NC.LoadedFiles	2	LAFITEFORMSMENU	6	NOTECARDTYPESTUBS	3		

PROPERTY INDEX

NOTECARDS	7
-----------------	---