

File created: 11-Dec-86 14:05:28 {POGO:PARC:XEROX}<LOOPSCORE>USERS>LOOPSIMAGEOBJECTS.;5

changes to: (VARS LOOPSIMAGEOBJECTSCOMS)

previous date: 19-Nov-86 10:06:04 {POGO:PARC:XEROX}<LOOPSCORE>USERS>LOOPSIMAGEOBJECTS.;4

Read Table: OLD-INTERLISP-FILE

Package: INTERLISP

Format: XCCS

(* * Copyright (c) 1985, 1986 by Xerox Corporation. All rights reserved.)

(RPAQQ **LOOPSIMAGEOBJECTSCOMS**

```
(( * * ImageObjects in LOOPS)
(* * Defines Loops wrappers around image objects.)
(FILE (FROM LISPUSERS)
      TEDIT)
(* Active value that enforces the connection between Lisp and Loops active values)
(CLASSES EnsureImageObject)
(METHODS EnsureImageObject.GetWrappedValue)
(* The basic Loops image object that provides the interface between the Lisp and Loops worlds.)
(FILE CacheObject)
(CLASSES LoopsImageObject)
(METHODS LoopsImageObject.AfterGetFromFile LoopsImageObject.AfterPutToFile
  LoopsImageObject.BeforePutToFile LoopsImageObject.ButtonEventIn LoopsImageObject.CachedImageBox
  LoopsImageObject.Copy LoopsImageObject.CopyButtonEventIn LoopsImageObject.Display
  LoopsImageObject.DisplayImageStream? LoopsImageObject.ImageBox LoopsImageObject.PrePrint
  LoopsImageObject.PrintText LoopsImageObject.WhenCopied LoopsImageObject.WhenDeleted
  LoopsImageObject.WhenInserted LoopsImageObject.WhenMoved LoopsImageObject.WhenOperatedOn)
(* These functions catch the LISP IMAGEOBJECT messages and pass them off to the Loops image object)
(FNS LoopsImageObjectButtonEventInFn LoopsImageObjectCopyButtonEventInFn LoopsImageObjectCopyFn
  LoopsImageObjectDisplayFn LoopsImageObjectGetFn LoopsImageObjectImageBoxFn
  LoopsImageObjectPrePrintFn LoopsImageObjectPutFn LoopsImageObjectWhenCopiedFn
  LoopsImageObjectWhenDeletedFn LoopsImageObjectWhenInsertedFn LoopsImageObjectWhenMovedFn
  LoopsImageObjectWhenOperatedOnFn)
[VARS (LoopsImageFns (IMAGEFNSCREATE (FUNCTION LoopsImageObjectDisplayFn)
  (FUNCTION LoopsImageObjectImageBoxFn)
  (FUNCTION LoopsImageObjectPutFn)
  (FUNCTION LoopsImageObjectGetFn)
  (FUNCTION LoopsImageObjectCopyFn)
  (FUNCTION LoopsImageObjectButtonEventInFn)
  (FUNCTION LoopsImageObjectCopyButtonEventInFn)
  (FUNCTION LoopsImageObjectWhenMovedFn)
  (FUNCTION LoopsImageObjectWhenInsertedFn)
  (FUNCTION LoopsImageObjectWhenDeletedFn)
  (FUNCTION LoopsImageObjectWhenCopiedFn)
  (FUNCTION LoopsImageObjectWhenOperatedOnFn)
  (FUNCTION LoopsImageObjectPrePrintFn)

(GLOBALVARS LoopsImageFns)
(* * Hooks that give users an easy way to insert LoopsImageObjects into a TEdit document. The function
  LIO is the main interface %. LIO is designed to be called inside of a TEDIT session, via the
  control-O image-object hook. Alternatly, META-W (or some other user-specifiable character)
  will call LIO directly.)
(FNS LIO InsertLoopsImageObject)
[INITVARS (LIOInsertCharCodes (LIST (CHARCODE ##W)
  (CHARCODE ##w)

[P (for char in LIOInsertCharCodes do (TEDIT.SETFUNCTION char (FUNCTION InsertLoopsImageObject)
(* * Definitions for the specialized Loops image object classes)
(COMS (* * Image objects that should only be used inside a TEdit document)
  (CLASSES TEditImageObject)
  (METHODS TEditImageObject.AllObjects TEditImageObject.CurrentCharLooks
    TEditImageObject.CurrentFont TEditImageObject.TEditIV TEditImageObject.TextStream))
(COMS (* * A wrapper for Lisp image object that converts them to Loops image objects)
  (CLASSES ImageObjectWrapper)
  (METHODS ImageObjectWrapper.AfterGetFromFile ImageObjectWrapper.AfterPutToFile
    ImageObjectWrapper.ButtonEventIn ImageObjectWrapper.Copy
    ImageObjectWrapper.CopyButtonEventIn ImageObjectWrapper.Display
    ImageObjectWrapper.GetIVValue ImageObjectWrapper.ImageBox ImageObjectWrapper.PrePrint
    ImageObjectWrapper.WhenCopied ImageObjectWrapper.WhenDeleted
    ImageObjectWrapper.WhenInserted ImageObjectWrapper.WhenMoved
    ImageObjectWrapper.WhenOperatedOn)
  (P (PUTHASH (QUOTE IMAGEOBJ)
    ($ ImageObjectWrapper)
    LispClassTable)))
(COMS (* * Image objects that have slots that can be edited)
  (CLASSES EditableImageObjectMixin)
  (METHODS EditableImageObjectMixin.ButtonEventIn EditableImageObjectMixin.DEditObject
    EditableImageObjectMixin.InstallEditSource EditableImageObjectMixin.MakeEditSource
    EditableImageObjectMixin.NewInstance))
(COMS (* * Image objects that show up as normally printed text)
  (* Depends on these other files)
  (CLASSES LabelImageMixin)
  (METHODS LabelImageMixin.Display LabelImageMixin.GetLabel LabelImageMixin.ImageBox
    LabelImageMixin.MakeEditSource))
(COMS (* * Image object that have an IV text that can be TEdited)
```

```

(CLASSES TEditableImageObjectMixin)
(METHODS TEditableImageObjectMixin.ButtonEventIn)
(COMS (* * Image objects that have side-effects at Hardcopy time)
(CLASSES HardcopySideEffectObject)
(METHODS HardcopySideEffectObject.Display HardcopySideEffectObject.GetDisplayLabel
HardcopySideEffectObject.GetHardcopyLabel HardcopySideEffectObject.GetLabel
HardcopySideEffectObject.HardcopySideEffect HardcopySideEffectObject.ImageBox)
(INITVARS (HardcopySideEffectObjectVisible? T))
(GLOBALVARS HardcopySideEffectObjectVisible?))
(COMS (* * Objects for creating an index with page numbers)
(CLASSES InitIndex IndexEntry CollectIndex)
(METHODS InitIndex.GetDisplayLabel InitIndex.HardcopySideEffect IndexEntry.Display
IndexEntry.GetDisplayLabel IndexEntry.GetHardcopyLabel IndexEntry.HardcopySideEffect
CollectIndex.GetDisplayLabel CollectIndex.HardcopySideEffect))
(COMS (* * Objects that let you refer to the page number of some other item in a document)
(CLASSES PageNote PageReference)
(METHODS PageNote.GetDisplayLabel PageNote.HardcopySideEffect PageReference.GetDisplayLabel
PageReference.GetHardcopyLabel))
(COMS (* * Objects for creating an TOC with section and page numbers)
(CLASSES InitTOC SectionHeading CollectTOC)
(METHODS CollectTOC.GetDisplayLabel CollectTOC.HardcopySideEffect InitTOC.GetDisplayLabel
InitTOC.HardcopySideEffect SectionHeading.ComputeSectionNumber SectionHeading.Display
SectionHeading.GetDisplayLabel SectionHeading.GetHardcopyLabel
SectionHeading.HardcopySideEffect))
(COMS (* * Objects that let you refer to the section number of some other item in the document)
(CLASSES SectionNote SectionReference)
(METHODS SectionNote.GetDisplayLabel SectionNote.HardcopySideEffect
SectionReference.GetDisplayLabel SectionReference.GetHardcopyLabel))
(COMS (* * Image object for displaying pretty-printed LISP forms)
(CLASSES PPIImageObject)
(METHODS PPIImageObject.ButtonEventIn PPIImageObject.Display PPIImageObject.ImageBox
PPIImageObject.PPForm)
(COMS (* * This is a patch -- why isn't this fn defined in the system?)
(FNS \DSPSPACEFACTOR.DISPLAY)))
(COMS (* * Image object for displaying a box around another image object)
(CLASSES BoxedImageObject)
(METHODS BoxedImageObject.ButtonEventIn BoxedImageObject.Display BoxedImageObject.ImageBox))
(COMS (* * Image objects that show the last time/place saved)
(CLASSES WhenSavedImageObject WhenLastSaved WhereLastSaved)
(METHODS WhenSavedImageObject.BeforePutToFile WhenSavedImageObject.Display
WhenSavedImageObject.GetLabel))
(COMS (* * Image objects or displaying text in tabular columns)
(CLASSES TableTextObject)
(METHODS TableTextObject.Display TableTextObject.ImageBox)))

```

(* * ImageObjects in LOOPS)

(* * Defines Loops wrappers around image objects.)

```

(FILESLoad (FROM LISPUSERS)
TEDIT)

```

(* * Active value that enforces the connection between Lisp and Loops active values)

```

(DEFCLASSES EnsureImageObject)

```

```

(DEFCLASS EnsureImageObject (MetaClass Class Edited: (* sml "27-Dec-85 15:56"))
(Supers FirstFetchAV))

```

```

(\BatchMethodDefs)

```

```

(METH EnsureImageObject GetWrappedValue (containingObj varName propName type)
(* Specialization)
)

```

```

(DEFINEQ

```

(EnsureImageObject.GetWrappedValue

```

(Method ((EnsureImageObject GetWrappedValue)
self containingObj varName propName type)

```

(* sml " 4-Oct-85 18:48")

(* Specialization)

```

(ReplaceActiveValue self (IMAGEOBJECTCREATE containingObj
LoopsImageFns)
containingObj varName propName type)))

```

```

)

```

```

(\UnbatchMethodDefs)

```

(* * The basic Loops image object that provides the interface between the Lisp and Loops worlds.)

```

(FILESLoad CacheObject)

```

```

(DEFCLASSES LoopsImageObject)

```

```

[DEFCLASS LoopsImageObject (MetaClass AbstractClass doc
  (* A Loops IMAGEOBJECT. The IMAGEFNS for the corresponding Lisp IMAGEOBJECT send an appropriate message to
  the instance.)

    Edited: (* sml " 3-Jan-86 10:04")
    )
  (Supers ObjectWithCache)
  (InstanceVariables (imageObject #. (Defer (IMAGEOBJCREATE containingObj LoopsImageFns))
    DontSave
    (Value)
    doc (* the image object that directs operations back to self --
    its IMAGEFNS is LoopsImageFns)
  )

(\BatchMethodDefs)

(METH LoopsImageObject AfterGetFromFile (textStream)
  (* * Sent to read back in the object from a file. See the PutToFile method.)

  (category FileI/O))

(METH LoopsImageObject AfterPutToFile (fileStream)
  (* * Sent to save self on the file. See also GetFromFile GetFromFile.
  NOTE: -
  This gets sent after the instance gets dumped to the file, so it really doesn't need to do anything special.)

  (category FileI/O))

(METH LoopsImageObject BeforePutToFile (stream)
  (* * The object is about to be written to a file...)

  (category FileI/O))

(METH LoopsImageObject ButtonEventIn (windowStream selection relX relY window textStream button)
  (* * Sent when you press a mouse button inside the imageobject)

  (category MouseEvent))

(METH LoopsImageObject CachedImageBox (imageStream)
  NIL
  (category Display))

(METH LoopsImageObject Copy NIL
  (* * Sent during a Copy-Select operation to generate a copy of the IMAGEBOJECT)

  (category Copying))

(METH LoopsImageObject CopyButtonEventIn (windowStream)
  (* * Sent when you hit the copy-button while inside the IMAGEOBJECT)

  (category MouseEvent))

(METH LoopsImageObject Display (imageStream)
  (* * Sent to display self in the image stream)

  (category Display))

(METH LoopsImageObject DisplayImageStream? (stream)
  (* * Is the given stream a display image stream?)

  (category Misc))

(METH LoopsImageObject ImageBox (imageStream currentX rightMargin)
  (* * Sent to find out the size and position of the IMAGEOBJECT in the image stream.
  Returns an IMAGEBOX region, with fields XSIZE YSIZE YDESC XKERN)

  (category Display))

(METH LoopsImageObject PrePrint NIL
  (* * The PrePrint fn method for LoopsImageObjects. Used to specify what should be BKSYSBUFed into a window if this obj
  is COPY-SELECTED.)

  (category Copying))

```

```

{MEDLEY}<loops>obsolete>from1.1>old>LOOPSIMAGEOBJECTS.;1

(METH LoopsImageObject PrintText (imageStream text font)
  (* * Center print the text in the image objects box)
  (category Display))

(METH LoopsImageObject WhenCopied (targetWindowStream sourceTextStream targetTextStream)
  (* * Sent to self when TEDIT copies the object. See also the Copy method)
  (category WhenChanged))

(METH LoopsImageObject WhenDeleted (targetWindowStream sourceTextStream targetTextStream)
  (* * Sent when TEDIT deletes self from the stream)
  (category WhenChanged))

(METH LoopsImageObject WhenInserted (targetWindowStream sourceTextStream targetTextStream)
  (* * Sent when TEDIT inserts self in the stream)
  (category WhenChanged))

(METH LoopsImageObject WhenMoved (targetWindowStream sourceTextStream targetTextStream)
  (* * Sent when TEDIT moves the object)
  (category WhenChanged))

(METH LoopsImageObject WhenOperatedOn (windowStream howOperatedOn selection textStream)
  (* * Sent when TEDIT does an edit operation on the object. Possible values of howOperatedOn are SELECTED,
  DESELECTED, HIGHLIGHTED, and UNHIGHLIGHTED)
  (category WhenChanged))

(DEFINEQ

(LoopsImageObject.AfterGetFromFile
  (Method ((LoopsImageObject AfterGetFromFile)
    self textStream)
    (* smL "10-Mar-86 14:23")
    (* * Sent to read back in the object from a file. See the PutToFile method.)
    (@ imageObject)))

(LoopsImageObject.AfterPutToFile
  (Method ((LoopsImageObject AfterPutToFile)
    self fileStream)
    (* smL "10-Mar-86 14:08")
    (* * Sent to save self on the file. See also GetFromFile GetFromFile.
    -
    NOTE: -
    This gets sent after the instance gets dumped to the file, so it really doesn't need to do anything special.)
    NIL))

(LoopsImageObject.BeforePutToFile
  (Method ((LoopsImageObject BeforePutToFile)
    self stream)
    (* smL: "31-Oct-85 15:58")
    (* * The object is about to be written to a file...)
    self))

(LoopsImageObject.ButtonEventIn
  (Method ((LoopsImageObject ButtonEventIn)
    self windowStream selection relX relY window textStream button)
    (* smL "13-Nov-86 17:52")
    (* * Sent when you press a mouse button inside the imageobject)

    (SELECTQ (MENU
      (create
        MENU
        ITEMS _
        [QUOTE
          ((SaveValue (QUOTE
            SaveValue
              )
            "Put this Lisp
            object in
            (SavedValue) "

```

```

(SUBITEMS
  SaveValue
  Inspect]
  CENTERFLG _ T))
(SaveValue (PutSavedValue self)
  NIL)
(Inspect (INSPECT self)
  NIL)
NIL)))

```

(LoopsImageObject.CachedImageBox

```

(Method ((LoopsImageObject CachedImageBox)
  self imageStream)
  (* sml "2-Jan-86 12:47")
  (* New method template)

  (OR (IMAGEOBJPROP (@ imageObject)
    (QUOTE BOUNDBOX))
    (_ self ImageBox imageStream))))

```

(LoopsImageObject.Copy

```

(Method ((LoopsImageObject Copy)
  self)
  (* sml "9-Jan-86 10:42")

  (* * Sent during a Copy-Select operation to generate a copy of the IMAGEOBJECT)

  (LET ((copyOfSelf (_ self CopyDeep))
    (_ copyOfSelf ClearCache)
    (change (@ copyOfSelf imageObject)
      (IMAGEOBJCREATE copyOfSelf LoopsImageFns)))))

```

(LoopsImageObject.CopyButtonEventIn

```

(Method ((LoopsImageObject CopyButtonEventIn)
  self windowStream)
  (* sml "14-Oct-85 11:33")

  (* * Sent when you hit the copy-button while inside the IMAGEOBJECT)

  NIL))

```

(LoopsImageObject.Display

```

(Method ((LoopsImageObject Display)
  self imageStream)
  (* sml "14-Oct-85 11:34")

  (* * Sent to display self in the image stream)

  NIL))

```

(LoopsImageObject.DisplayImageStream?

```

(Method ((LoopsImageObject DisplayImageStream?)
  self stream)
  (* sml "14-Oct-85 12:58")

  (* * Is the given stream a display image stream?)

  [AND (IMAGESTREAMP stream)
    (LET ((displayType (IMAGESTREAMTYPE stream)))
      (OR (EQ displayType (QUOTE DISPLAY))
        (EQUAL displayType (QUOTE (COLOR DISPLAY)))]))

```

(LoopsImageObject.ImageBox

```

(Method ((LoopsImageObject ImageBox)
  self imageStream currentX rightMargin)
  (* sml "14-Oct-85 11:34")

  (* * Sent to find out the size and position of the IMAGEOBJECT in the image stream.
  Returns an IMAGEBOX region, with fields XSIZE YSIZE YDESC XKERN)

  (create IMAGEBOX)))

```

(LoopsImageObject.PrePrint

```

(Method ((LoopsImageObject PrePrint)
  self)
  (* sml "7-Mar-86 18:49")

  (* * The PrePrint fn method for LoopsImageObjects. Used to specify what should be BKSYSEBUFe into a window if this obj
  is COPY-SELECTED.)

  self))

```

(LoopsImageObject.PrintText

```

(Method ((LoopsImageObject PrintText)
  self imageStream text font)
  (* sml "7-Mar-86 19:06")

  (* * Center print the text in the image objects box)

```

```

(RESETLST
  (if font
    then (RESETSAVE (DSPFONT font imageStream)
      (LIST (FUNCTION DSPFONT)
        (DSPFONT NIL imageStream)
        imageStream)))
  (LET ((imageBox (_ self CachedImageBox imageStream))
    (xPos (DSPXPOSITION NIL imageStream))
    (yPos (DSPYPOSITION NIL imageStream))
    (CENTERPRINTINREGION text (CREATEREGION (DIFFERENCE xPos
      (fetch XKERN of imageBox))
      (DIFFERENCE yPos
        (fetch YDESC of imageBox))
        )
      (fetch XSIZE of imageBox)
      (fetch YSIZE of imageBox))
      imageStream)
    (MOVE TO (PLUS xPos (fetch XSIZE of imageBox))
      (PLUS yPos (fetch YSIZE of imageBox))
      imageStream))))

```

(LoopsImageObject.WhenCopied

```

(Method ((LoopsImageObject WhenCopied)
  self targetWindowStream sourceTextStream targetTextStream)
  (* sml "14-Oct-85 11:34")

(* * Sent to self when TEDIT copies the object. See also the Copy method)

NIL))

```

(LoopsImageObject.WhenDeleted

```

(Method ((LoopsImageObject WhenDeleted)
  self targetWindowStream sourceTextStream targetTextStream)
  (* sml "14-Oct-85 11:35")

(* * Sent when TEDIT deletes self from the stream)

NIL))

```

(LoopsImageObject.WhenInserted

```

(Method ((LoopsImageObject WhenInserted)
  self targetWindowStream sourceTextStream targetTextStream)
  (* sml "14-Oct-85 11:35")

(* * Sent when TEDIT inserts self in the stream)

NIL))

```

(LoopsImageObject.WhenMoved

```

(Method ((LoopsImageObject WhenMoved)
  self targetWindowStream sourceTextStream targetTextStream)
  (* sml "14-Oct-85 11:35")

(* * Sent when TEDIT moves the object)

NIL))

```

(LoopsImageObject.WhenOperatedOn

```

(Method ((LoopsImageObject WhenOperatedOn)
  self windowStream howOperatedOn selection textStream)
  (* sml "14-Oct-85 11:35")

(* * Sent when TEDIT does an edit operation on the object. Possible values of howOperatedOn are SELECTED,
DESELECTED, HIGHLIGHTED, and UNHIGHLIGHTED)

NIL))

```

)

(\UnbatchMethodDefs)

(* * These functions catch the LISP IMAGEOBJECT messages and pass them off to the Loops image object)

(DEFINEQ

(LoopsImageObject.ButtonEventInFn

```

[LAMBDA (imageObject windowStream selection relX relY window textStream button)
  (* sml "25-Sep-85 17:13")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))

```

```
ButtonEventIn windowStream selection relX relY window textStream button])
```

(LoopsImageObjectCopyButtonEventInFn

```
[LAMBDA (imageObject windowStream) (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    CopyButtonEventIn windowStream])
```

(LoopsImageObjectCopyFn

```
[LAMBDA (imageObject) (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    Copy)]
```

(LoopsImageObjectDisplayFn

```
[LAMBDA (imageObject imageStream) (* sml "25-Sep-85 17:14")
  (* Comment)
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    Display imageStream)]
```

(LoopsImageObjectGetFn

```
[LAMBDA (textStream) (* sml "10-Mar-86 14:25")
  (LET [(self (EVAL (READ textStream FILERDTBL)
    (_ self AfterGetFromFile textStream)))]
```

(LoopsImageObjectImageBoxFn

```
[LAMBDA (imageObject imageStream currentX rightMargin) (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    ImageBox imageStream currentX rightMargin)]
```

(LoopsImageObjectPrePrintFn

```
[LAMBDA (imageObject) (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    PrePrint)]
```

(LoopsImageObjectPutFn

```
[LAMBDA (imageObject fileStream) (* sml "10-Mar-86 14:22")
```

(** This is the PUTFN for Loops image objects. -

The only trick here is that the PutToFile msg for an object needn't really do anything, as this function dumps the instance. Why? Well, when reading it back in, the instance needs to exist before it can be sent a GetFromFile msg, so the GETFN function must be able to read the instance from the TEDIT file, so it needs to know what the format of the instance is...)

```
  (LET [(self (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM)
    (_ self BeforePutToFile fileStream)
    (PRIN2 (_ self MakeFileSource)
      fileStream)
    (_ self AfterPutToFile fileStream)))]
```

(LoopsImageObjectWhenCopiedFn

```
[LAMBDA (imageObject targetWindowStream sourceTextStream targetTextStream)
  (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    WhenCopied targetWindowStream sourceTextStream targetTextStream)]
```

(LoopsImageObjectWhenDeletedFn

```
[LAMBDA (imageObject targetWindowStream sourceTextStream targetTextStream)
  (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    WhenDeleted targetWindowStream sourceTextStream targetTextStream)]
```

(LoopsImageObjectWhenInsertedFn

```
[LAMBDA (imageObject targetWindowStream sourceTextStream targetTextStream)
  (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    WhenInserted targetWindowStream sourceTextStream targetTextStream)]
```

(LoopsImageObjectWhenMovedFn

```
[LAMBDA (imageObject targetWindowStream sourceTextStream targetTextStream)
  (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
    WhenMoved targetWindowStream sourceTextStream targetTextStream)]
```

(LoopsImageObjectWhenOperatedOnFn

```
[LAMBDA (imageObject windowStream howOperatedOn selection textStream)
  (* sml "25-Sep-85 14:18")
  (_ (IMAGEOBJPROP imageObject (QUOTE OBJECTDATUM))
```

```

    WhenOperatedOn windowStream howOperatedOn selection textStream])
)

```

```

(RPAQ LoopsImageFns
  (IMAGEFNSCREATE (FUNCTION LoopsImageObjectDisplayFn)
    (FUNCTION LoopsImageObjectImageBoxFn)
    (FUNCTION LoopsImageObjectPutFn)
    (FUNCTION LoopsImageObjectGetFn)
    (FUNCTION LoopsImageObjectCopyFn)
    (FUNCTION LoopsImageObjectButtonEventInFn)
    (FUNCTION LoopsImageObjectCopyButtonEventInFn)
    (FUNCTION LoopsImageObjectWhenMovedFn)
    (FUNCTION LoopsImageObjectWhenInsertedFn)
    (FUNCTION LoopsImageObjectWhenDeletedFn)
    (FUNCTION LoopsImageObjectWhenCopiedFn)
    (FUNCTION LoopsImageObjectWhenOperatedOnFn)
    (FUNCTION LoopsImageObjectPrePrintFn)))
)

```

```

(DECLARE: DOEVAL@COMPILE DONTCOPY

```

```

(GLOBALVARS LoopsImageFns)
)

```

(* * Hooks that give users an easy way to insert LoopsImageObjects into a TEdit document.
The function LIO is the main interface %. LIO is designed to be called inside of a TEDIT session, via the control-O image-object hook. Alternatly, META-W (or some other user-specifiable character) will call LIO directly.)

```

(DEFINEQ

```

```

(LIO
  [LAMBDA NIL
    (* sml " 2-Jan-86 09:28")

    (* * Creates a new LoopsImageObject, returning the wrapped Lisp image object.
    Intended to be used inside of TEDIT.)

    (LET [(choice (MENU (create MENU
      TITLE _ "Image object class"
      CENTERFLG _ T
      ITEMS _ [SORT (for class in (_ ($ LoopsImageObject)
        List!
        (QUOTE Subclasses))
        when (NOT (_ ($! class)
          InstOf!
          (QUOTE AbstractClass)))
        collect (LIST class ($! class)))
      (FUNCTION (LAMBDA (x y)
        (ALPHORDER (CAR x)
          (CAR y)
          (FUNCTION (LAMBDA (item menu button)
            (PROMPTPRINT (GetClassHere (CADR item)
              (QUOTE doc]
            (if choice
              then (@ (_ choice New)
                imageObject)
              else NIL])
    ]

```

```

(InsertLoopsImageObject
  [LAMBDA (textStream textObj)
    (* sml " 2-Jan-86 10:23")

```

(* * Intended to be bound to a meta key in the TEDIT readtable, so that #W (say) will insert a Loops image object)

```

  (LET ((obj (LIO)))
    (if obj
      then (TEDIT.INSERT.OBJECT obj textStream])
  )

```

```

(RPAQ? LIOInsertCharCodes (LIST (CHARCODE #W)
  (CHARCODE #w)))

```

```

(for char in LIOInsertCharCodes do (TEDIT.SETFUNCTION char (FUNCTION InsertLoopsImageObject)))

```

(* * Definitions for the specialized Loops image object classes)

(* * Image objects that should only be used inside a TEdit document)

```

(DEFCLASSES TEditImageObject)

```

```

(DEFCLASS TEditImageObject (MetaClass AbstractClass Edited:
  (Supers LoopsImageObject))
  (* sml " 7-Mar-86 18:36")

```

```

(\BatchMethodDefs)

```



```

(METH TEditImageObject AllObjects (textStream)
  ;; Return a list of all the ImageObjects contained in the given stream, in order
  (category (TEditImageObject)))

(METH TEditImageObject CurrentCharLooks (imageStream)
  NIL)

(METH TEditImageObject CurrentFont (imageStream)
  ) (* New method template)

(METH TEditImageObject TEditIV (ivName textStream)
  (* * TEdit an IV of this image object, and when done notify the TEdit stream containing this image object that something has
  chagned.)
  )

(METH TEditImageObject TextStream (imageStream)
  NIL)

(DEFINEQ
  (TEditImageObject.AllObjects
    (Method ((TEditImageObject AllObjects)
      self textStream)
      (* smL "20-Oct-86 18:29")
      ;; Return a list of all the ImageObjects contained in the given stream, in order
      ;;
      ;; This code is lifted from the NoteCards function TEDIT.LIST.OF.OBJECTS, originally written by RBB.
      ;;
      (LET ((OBJLIST (TCONC NIL)))
        (DECLARE (SPECVARS OBJLIST))
        (TEDIT.MAPPICES (TEXTOBJ textStream)
          [FUNCTION (LAMBDA (CH# PC PC# OBL)
            (COND
              ((AND PC (NEQ PC (QUOTE LASTPIECE))
                (fetch POBJ of PC))
               ; If there is an imageobj in this piece then add it to the list.
               (TCONC OBL (fetch POBJ of PC]
              (OBJLIST)
              (CDAR OBJLIST))))))

  (TEditImageObject.CurrentCharLooks
    (Method ((TEditImageObject CurrentCharLooks)
      self imageStream)
      (* smL " 7-Mar-86 18:46")
      (* * If displaying in a TEDIT window, return the current font for this imageObject, Otherwise, return NIL.)
      [LET ((textStream (_ self TextStream imageStream)))
        (if textStream
          then (TEDIT.GET.LOOKS textStream (GETFILEPTR textStream))

  (TEditImageObject.CurrentFont
    (Method ((TEditImageObject CurrentFont)
      self imageStream)
      (* smL " 7-Mar-86 18:44")
      (* New method template)
      (LET ((looks (_ self CurrentCharLooks imageStream)))
        (if looks
          then (FONTCREATE (LISTGET looks (QUOTE FAMILY))
            (LISTGET looks (QUOTE SIZE))
            (LIST (LISTGET looks (QUOTE WEIGHT))
              (LISTGET looks (QUOTE SLOPE))
              (LISTGET looks (QUOTE EXPANSION)))
            0
            (IMAGESTREAMTYPE imageStream))
          else (DSPFONT NIL imageStream)))))

  (TEditImageObject.TEditIV
    (Method ((TEditImageObject TEditIV)
      self ivName textStream)
      (* smL " 7-Mar-86 18:44")
      (* * TEdit an IV of this image object, and when done notify the TEdit stream containing this image object that something has
      chagned.)
      (ADD.PROCESS (LIST [FUNCTION (LAMBDA (self ivName textStream)
        (change (GetValue self ivName)
          (TEDIT (MKSTRING DATUM)
            NIL T))
        (if (TEXTSTREAMP textStream)
          then (TEDIT.OBJECT.CHANGED textStream
            (@ imageObject]

```

```

(KWOTE self)
(KWOTE ivName)
(KWOTE textStream))
(QUOTE NAME)
(CONCAT (QUOTE TeditIV-
ivName))))

```

(TeditImageObject.TextStream

```

(Method ((TeditImageObject TextStream)
self imageStream)
(* sml "10-Mar-86 13:48")

(* * What is the textstream containing this image object?)

(DECLARE (SPECVARS TEXTOBJ TEXTSTREAM))
(if (AND (BOUNDP (QUOTE TEXTSTREAM))
(TEXTSTREAMP TEXTSTREAM))
then TEXTSTREAM
elseif (AND (BOUNDP (QUOTE TEXTOBJ))
(type? TEXTOBJ TEXTOBJ))
then (TEXTSTREAM TEXTOBJ)
else (HELP "Can't find the textstream containing self!"))))
)

(\UnbatchMethodDefs)

```

(* * A wrapper for Lisp image object that converts them to Loops image objects)

```

(DEFCLASSES ImageObjectWrapper)

(DEFCLASS ImageObjectWrapper (MetaClass AbstractClass doc
Edited:
)
(* Provides a class for Lisp IMAGEOBJECTs via the
LoopsDateType hook.)
(* sml " 7-May-86 17:20")

(Supers LoopsImageObject))

(\BatchMethodDefs)

(METH ImageObjectWrapper AfterGetFromFile (textStream)
)
(* New method template)

(METH ImageObjectWrapper AfterPutToFile (fileStream)
)
(* New method template)

(METH ImageObjectWrapper ButtonEventIn (windowStream selection relX relY window textStream button)
)
(* New method template)

(METH ImageObjectWrapper Copy NIL
)
(* New method template)

(METH ImageObjectWrapper CopyButtonEventIn (windowStream)
)
(* New method template)

(METH ImageObjectWrapper Display (imageStream)
)
(* New method template)

(METH ImageObjectWrapper GetIVValue (varName propName)
(* * Get the value of an "iv" from self)
)

(METH ImageObjectWrapper ImageBox (imageStream currentX rightMargin)
)
(* New method template)

(METH ImageObjectWrapper PrePrint NIL
)
(* New method template)

(METH ImageObjectWrapper WhenCopied (targetWindowStream sourceTextStream targetTextStream)
)
(* New method template)

(METH ImageObjectWrapper WhenDeleted (targetWindowStream sourceTextStream targetTextStream)
)
(* New method template)

(METH ImageObjectWrapper WhenInserted (targetWindowStream sourceTextStream targetTextStream)
)
(* New method template)

(METH ImageObjectWrapper WhenMoved (targetWindowStream sourceTextStream targetTextStream)
)
(* New method template)

(METH ImageObjectWrapper WhenOperatedOn (windowStream howOperatedOn selection textStream)
)

```

(* New method template)

)

(DEFINEQ

(ImageObjectWrapper.AfterGetFromFile(Method ((ImageObjectWrapper AfterGetFromFile)
self textStream)

(* smL " 7-May-86 16:27")

(* New method template)

self))

(ImageObjectWrapper.AfterPutToFile(Method ((ImageObjectWrapper AfterPutToFile)
self fileStream)

(* smL " 7-May-86 16:28")

(* New method template)

(WRITEIMAGEOBJ self fileStream)))

(ImageObjectWrapper.ButtonEventIn(Method ((ImageObjectWrapper ButtonEventIn)
self windowStream selection relX relY window textStream button)

(* smL " 7-May-86 17:18")

(* New method template)

(APPLY* (IMAGEOBJPROP self
(QUOTE BUTTONEVENTINFN
))self windowStream selection
relX relY window textStream
button)))**(ImageObjectWrapper.Copy**(Method ((ImageObjectWrapper Copy)
self)

(* smL " 7-May-86 17:26")

(* New method template)

(APPLY* (IMAGEOBJPROP self (QUOTE COPYFN))
self)))**(ImageObjectWrapper.CopyButtonEventIn**(Method ((ImageObjectWrapper CopyButtonEventIn)
self windowStream)

(* smL " 7-May-86 17:09")

(* New method template)

(APPLY* (IMAGEOBJPROP self (QUOTE COPYBUTTONEVENTINFN))
self windowStream)))**(ImageObjectWrapper.Display**(Method ((ImageObjectWrapper Display)
self imageStream)

(* smL " 7-May-86 17:10")

(* New method template)

(_ self ImageObjectFn (QUOTE DisplayFn)
(LIST (@ wrappedImageObject)
imageStream))
(APPLY* (IMAGEOBJPROP self (QUOTE DISPLAYFN))
self imageStream)))**(ImageObjectWrapper.GetIVValue**(Method ((ImageObjectWrapper GetIVValue)
self varName propName)

(* smL " 7-May-86 16:23")

(* * Get the value of an "iv" from self)

(ERROR "Can't get the IV of a Lisp IMAGEOBJ" self)))

(ImageObjectWrapper.ImageBox(Method ((ImageObjectWrapper ImageBox)
self imageStream currentX rightMargin)

(* smL " 7-May-86 17:12")

(* New method template)

(APPLY* (IMAGEOBJPROP self (QUOTE IMAGEBOXFN))
self imageStream currentX rightMargin)))**(ImageObjectWrapper.PrePrint**(Method ((ImageObjectWrapper PrePrint)
self)

(* smL " 7-May-86 17:13")

(* New method template)

(APPLY* (IMAGEOBJPROP self (QUOTE PREPRINTFN))
self)))**(ImageObjectWrapper.WhenCopied**(Method ((ImageObjectWrapper WhenCopied)
self targetWindowStream sourceTextStream targetTextStream)

```

(* smL " 7-May-86 17:15")
(* New method template)
(APPLY* (IMAGEOBJPROP self (QUOTE
                                WHENCOPIEDFN
                                )))
self targetWindowStream
sourceTextStream targetTextStream))

```

(ImageObjectWrapper.WhenDeleted

```

(Method ((ImageObjectWrapper WhenDeleted)
self targetWindowStream sourceTextStream targetTextStream)
(* smL " 7-May-86 17:16")
(* New method template)
(APPLY* (IMAGEOBJPROP self (QUOTE
                                WHENDELETEDFN
                                )))
self targetWindowStream
sourceTextStream targetTextStream))

```

(ImageObjectWrapper.WhenInserted

```

(Method ((ImageObjectWrapper WhenInserted)
self targetWindowStream sourceTextStream targetTextStream)
(* smL " 7-May-86 17:17")
(* New method template)
(APPLY* (IMAGEOBJPROP self (QUOTE
                                WHENINSERTEDFN
                                )))
self targetWindowStream
sourceTextStream targetTextStream))

```

(ImageObjectWrapper.WhenMoved

```

(Method ((ImageObjectWrapper WhenMoved)
self targetWindowStream sourceTextStream targetTextStream)
(* smL " 7-May-86 17:18")
(* New method template)
(APPLY* (IMAGEOBJPROP self (QUOTE
                                WHENMOVEDFN
                                )))
self targetWindowStream
sourceTextStream targetTextStream))

```

(ImageObjectWrapper.WhenOperatedOn

```

(Method ((ImageObjectWrapper WhenOperatedOn)
self windowStream howOperatedOn selection textStream)
(* smL " 7-May-86 17:18")
(* New method template)
(APPLY* (IMAGEOBJPROP self (QUOTE
                                WHENOPERATEDONFN
                                )))
self windowStream howOperatedOn
selection textStream))

```

```

(\UnbatchMethodDefs)

```

```

(PUTHASH (QUOTE IMAGEOBJ)
($ ImageObjectWrapper)
LispClassTable)

```

```

(* * Image objects that have slots that can be edited)

```

```

(DEFCLASSES EditableImageObjectMixin)

```

```

(DEFCLASS EditableImageObjectMixin (MetaClass AbstractClass Edited:
(* smL " 7-Mar-86 19:02"))
(Supers Tofu))

```

```

(\BatchMethodDefs)

```

```

(METH EditableImageObjectMixin ButtonEventIn (windowStream selection relX rely window textStream button)

```

```

(* * Sent when you press a mouse button inside the imageobject)

```

```

(method EditableImageObjectMixin.ButtonEventIn ButtonEventIn #.NotSetValue Display* #.NotSetValue GetLabel
#. NotSetValue))

```

```

(METH EditableImageObjectMixin DEditObject (textStream)

```

```

NIL)

(METH EditableImageObjectMixin InstallEditSource (editedDescription)
      (* Specialization)
)

(METH EditableImageObjectMixin MakeEditSource NIL NIL)

(METH EditableImageObjectMixin NewInstance (name arg1 arg2 arg3 arg4 arg5)
      NIL)

(DEFINEQ
  (EditableImageObjectMixin.ButtonEventIn
    (Method ((EditableImageObjectMixin ButtonEventIn)
              self windowStream selection relX relY window textStream button)
              (* sml "17-Feb-86 16:50")

    (* * Sent when you press a mouse button inside the imageobject)

    (SELECTQ (MENU
              (create
                MENU
                ITEMS _
                (QUOTE
                  ((SaveValue (QUOTE
                              SaveValue
                              )
                              "Put this Lisp
                              object in
                              (SavedValue) "
                              (SUBITEMS
                                SaveValue
                                Inspect))
                              Edit))
                  CENTERFLG _ T))
              (SaveValue (_Proto
                          ($ LoopsIcon)
                          Save self)
                          NIL)
              (Inspect (INSPECT self)
                        NIL)
              (Edit (_ self DEditObject
                      textStream)
                      (LET
                        [(window
                          (if (WINDOWP
                              windowStream)
                              then windowStream
                              else
                                (for w
                                  in (OPENWINDOWS)
                                  thereis
                                    (EQ
                                      windowStream
                                      (WINDOWPROP
                                        w
                                        (QUOTE DSP]
                                      (if window
                                        then (GIVE.TTY.PROCESS
                                              window)))
                                      NIL)
                                NIL)
                        ]
                      NIL)))

    (EditableImageObjectMixin.DEditObject
      (Method ((EditableImageObjectMixin DEditObject)
                self textStream)
                (* sml " 3-Jan-86 14:46")
                (* New method template)
                (ADD.PROCESS (LIST [FUNCTION (LAMBDA (self textStream)
                                              (_ self Edit)
                                              (if (TEXTSTREAMP textStream)
                                                  then (EDIT.OBJECT.CHANGED textStream
                                                                              (@ imageObject]
                                                                              (KWOTE self)
                                                                              (KWOTE textStream))
                                                  (QUOTE NAME)
                                                  (QUOTE EditImageObj)))))

    (EditableImageObjectMixin.InstallEditSource
      (Method ((EditableImageObjectMixin InstallEditSource)
                self editedDescription)
                (* sml " 3-Jan-86 11:15")
                (* Specialization)
                (LET ((imageObj (@ imageObject)))

    (* This needs to be done because the MakeEditSource method striped out the imageObject IV, and the standard
    InstallEditSource blanks all IVs of the instance...))

```

```
(PROG1 (_Super)
  (change (@ imageObject)
    imageObj))))))
```

(EditableImageObjectMixin.MakeEditSource

```
(Method ((EditableImageObjectMixin MakeEditSource)
  self)
  (* sml "29-Dec-85 15:52")
  (* Specialization)

  (LET ((editSource (_Super)))
    (REMOVE (ASSOC (QUOTE imageObject)
      editSource)
      editSource))))
```

(EditableImageObjectMixin.NewInstance

```
(Method ((EditableImageObjectMixin NewInstance)
  self name arg1 arg2 arg3 arg4 arg5)
  (* sml "25-Dec-85 12:57")
  (* Specialization)

  (PROG1 (_Super)
    (_ self Edit))))
```

```
)
```

```
(\UnbatchMethodDefs)
```

```
(* * Image objects that show up as normally printed text)
```

```
(* * Depends on these other files)
```

```
(DEFCLASSES LabelImageMixin)
```

```
(DEFCLASS LabelImageMixin (MetaClass AbstractClass Edited:
  (Supers TEditImageObject))
  (* sml: "31-Oct-85 16:06"))
```

```
(\BatchMethodDefs)
```

```
(METH LabelImageMixin Display (imageStream)
  NIL)
```

```
(METH LabelImageMixin GetLabel (imageStream)
  )
  (* New method template)
```

```
(METH LabelImageMixin ImageBox (imageStream currentX rightMargin)
  NIL)
```

```
(METH LabelImageMixin MakeEditSource NIL
  (* Specialization))
```

```
(DEFINEQ
```

(LabelImageMixin.Display

```
(Method ((LabelImageMixin Display)
  self imageStream)
  (* sml "25-Dec-85 12:35")
  (* Specialization)

  (_ self PrintText imageStream (_ self GetCache (QUOTE label))
    (_ self GetCache (QUOTE font))))))
```

(LabelImageMixin.GetLabel

```
(Method ((LabelImageMixin GetLabel)
  self imageStream)
  (* sml "22-Dec-85 21:01")
  (* New method template)

  "No label yet"))
```

(LabelImageMixin.ImageBox

```
(Method ((LabelImageMixin ImageBox)
  self imageStream currentX rightMargin)
  (* sml "9-Jan-86 11:21")
  (* Specialization)

  (LET [(font (_ self PutCache (QUOTE font)
    (_ self CurrentFont imageStream)
    (create IMAGEBOX
      XSIZE _ (STRINGWIDTH (_ self PutCache (QUOTE label)
        (_ self GetLabel
          imageStream))
          font)
      YSIZE _ (FONTPROP font (QUOTE HEIGHT))
      YDESC _ (FONTPROP font (QUOTE DESCENT))
      XKERN _ 0)))]
```

(LabelImageMixin.MakeEditSource

```
(Method ((LabelImageMixin MakeEditSource)
  self)
  (* sml "29-Dec-85 15:54"))
```

(* Specialization)

```
(LET ((editSource (_Super)))
  (REMOVE (ASSOC (QUOTE cache)
                 editSource)
           editSource)))
```

)

(\UnbatchMethodDefs)

(* * Image object that have an IV text that can be TEdited)

(DEFCLASSES TEditableImageObjectMixin)

```
(DEFCLASS TEditableImageObjectMixin (MetaClass AbstractClass Edited:
  (* sml " 7-Mar-86 19:03"))
```

```
(Supers EditableImageObjectMixin)
(InstanceVariables (text "" doc
```

(* the text of the image object. This can be TEdited)])

(\BatchMethodDefs)

(METH TEditableImageObjectMixin ButtonEventIn (windowStream selection relX rely window textStream button)

(* * Sent when you press a mouse button inside the imageobject)

```
(method TEditableImageObjectMixin.ButtonEventIn ButtonEventIn #.NotSetValue Display* #.NotSetValue
  GetLabel #. NotSetValue))
```

(DEFINEQ

(TEditableImageObjectMixin.ButtonEventIn

```
(Method ((TEditableImageObjectMixin ButtonEventIn)
  self windowStream selection relX rely window textStream button)
```

(* sml " 3-Jan-86 15:59")

(* * Sent when you press a mouse button inside the imageobject)

```
(SELECTQ (MENU
  (create
    MENU
    ITEMS _
    (QUOTE
      ((SaveValue (QUOTE
                    SaveValue
                    )
                  "Put this Lisp
                  object in
                  (SavedValue)"
                  (SUBITEMS
                    SaveValue
                    Inspect))
            Edit EditText))
    CENTERFLG _ T))
  (SaveValue (_Proto
              ($ LoopsIcon)
              Save self)
    NIL)
  (Inspect (INSPECT self)
    NIL)
  (Edit (_ self DEditObject
          textStream)
    NIL)
  (EditText (_ self TEditIV
                (QUOTE text)
                textStream)
    NIL)
  NIL)))
```

)

(\UnbatchMethodDefs)

(* * Image objects that have side-effects at Hardcopy time)

(DEFCLASSES HardcopySideEffectObject)

```
(DEFCLASS HardcopySideEffectObject (MetaClass AbstractClass Edited:
  (* sml " 3-Jan-86 10:36"))
```

(Supers LabelImageMixin))

(\BatchMethodDefs)

```
(METH HardcopySideEffectObject Display (imageStream)
  NIL)
```

```
(METH HardcopySideEffectObject GetDisplayLabel (imageStream)      (* New method template)
)

(METH HardcopySideEffectObject GetHardcopyLabel (imageStream)
NIL)

(METH HardcopySideEffectObject GetLabel (imageStream)              (* New method template)
)

(METH HardcopySideEffectObject HardcopySideEffect (imageStream)
NIL)

(METH HardcopySideEffectObject ImageBox (imageStream currentX rightMargin)
NIL)

(DEFINEQ
```

(HardcopySideEffectObject.Display

```
(Method ((HardcopySideEffectObject Display)
self imageStream)
(* sml "3-Jan-86 10:43")
(* Specialization)
```

```
[LET ((xPos (DSPXPOSITION NIL imageStream))
(yPos (DSPYPOSITION NIL imageStream))
(imageBox (_ self CachedImageBox imageStream)))
(PROG1 (_Super)
(if (_ self DisplayImageStream? imageStream)
then
```

(* Draw a box around the object, then invert the whole thing. The box will then show up if the object is inverted again, like it is when it is selected for deletion.)

```
(LET ((x0 (DIFFERENCE xPos (fetch XKERN of imageBox)))
(y0 (DIFFERENCE yPos (fetch YDESC of imageBox)))
(xSize (fetch XSIZE of imageBox))
(ySize (fetch YSIZE of imageBox)))
(BITBLT NIL NIL NIL imageStream x0 y0 1 ySize (QUOTE TEXTURE
)
(QUOTE PAINT)
BLACKSHADE)
(BITBLT NIL NIL NIL imageStream (PLUS x0 (SUB1 xSize))
y0 1 ySize (QUOTE TEXTURE)
(QUOTE PAINT)
BLACKSHADE)
(BITBLT NIL NIL NIL imageStream x0 y0 xSize 1 (QUOTE TEXTURE
)
(QUOTE PAINT)
BLACKSHADE)
(BITBLT NIL NIL NIL imageStream x0 (PLUS y0 (SUB1 ySize))
xSize 1 (QUOTE TEXTURE)
(QUOTE PAINT)
BLACKSHADE)
(BITBLT NIL NIL NIL imageStream x0 y0 xSize ySize
(QUOTE TEXTURE)
(QUOTE INVERT)
BLACKSHADE))))))
```

(HardcopySideEffectObject.GetDisplayLabel

```
(Method ((HardcopySideEffectObject GetDisplayLabel)
self imageStream)
(* sml "25-Dec-85 12:50")
(* New method template)

"Hardcopy side effect"))
```

(HardcopySideEffectObject.GetHardcopyLabel

```
(Method ((HardcopySideEffectObject GetHardcopyLabel)
self imageStream)
(* sml "25-Dec-85 12:51")
(* New method template)

""))
```

(HardcopySideEffectObject.GetLabel

```
(Method ((HardcopySideEffectObject GetLabel)
self imageStream)
(* sml "26-Dec-85 15:26")
(* New method template)

(if (_ self DisplayImageStream? imageStream)
then (if HardcopySideEffectObjectVisible?
then (_ self GetDisplayLabel imageStream)
else "")
else (_ self GetHardcopyLabel imageStream)))
```

(HardcopySideEffectObject.HardcopySideEffect

```
(Method ((HardcopySideEffectObject HardcopySideEffect)
self imageStream)
(* sml "25-Dec-85 12:47")
```

(* * Called at hardcopy time to perform any desired side-effects)


```
self))

(HardcopySideEffectObject.ImageBox
 (Method ((HardcopySideEffectObject ImageBox)
  self imageStream currentX rightMargin)
  (* sml " 2-Jan-86 13:30")
  (* Specialization)
  (if (NOT (_ self DisplayImageStream? imageStream))
    then (_ self HardcopySideEffect imageStream)
    (_Super)
  else (LET ((imageBox (_Super)))
    (add (fetch XSIZE of imageBox)
      2)
    (add (fetch YSIZE of imageBox)
      2)
    imageBox)))
)

(\UnbatchMethodDefs)

(RPAQ? HardcopySideEffectObject.Visible? T)

(DECLARE: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS HardcopySideEffectObject.Visible?)
)

(* * Objects for creating an index with page numbers)

(DEFCLASSES InitIndex IndexEntry CollectIndex)

(DEFCLASS InitIndex (MetaClass Class doc
  Edited:
  )
  (* Initializes the index. Must occur before the first IndexEntry in
  the TEdit file.)
  (* sml " 3-Jan-86 10:39")
  (Supers HardcopySideEffectObject))

[DEFCLASS IndexEntry (MetaClass Class doc
  Edited:
  )
  (* Marks this page number for inclusion in the index)
  (* sml " 2-Jan-86 14:06")
  (Supers TEditableImageObjectMixin HardcopySideEffectObject)
  (InstanceVariables (text "" doc
    (displayText? T doc
      (* the string for the index)
      (* should the text be included in the document?))
    )
  )]

[DEFCLASS CollectIndex (MetaClass Class doc
  Edited:
  )
  (* Collects up all the index marks in a document and puts them in a seperate TEdit stream)
  (* sml " 3-Jan-86 10:39")
  (Supers EditableImageObjectMixin HardcopySideEffectObject)
  (InstanceVariables (looks (FAMILY HELVETICA SIZE 10)
    doc
    )
    (* the looks for the index)
    (paraLooks (TABS (NIL (216 . DOTTEDRIGHT))
      LINELEADING 0 POSTPARALEADING 0 PARALEADING 1 RIGHTMARGIN 0 LEFTMARGIN 120
      1STLEFTMARGIN 0 QUAD LEFT)
      (* the para looks)
      doc
      )
    [firstPageFormat (TEDIT.SINGLE.PAGEFORMAT T 25.5 3.0 (@ looks)
      (QUOTE CENTERED)
      6 6 6 6 2 NIL 12 NIL (QUOTE PICAS)
      (QUOTE (FOLIOINFO (ARABIC "Index-" "-"))
    ]
    [versoPageFormat (TEDIT.SINGLE.PAGEFORMAT T 25.5 3.0 (@ looks)
      (QUOTE CENTERED)
      6 6 6 6 2 NIL 12 NIL (QUOTE PICAS)
      (QUOTE (FOLIOINFO (ARABIC "Index-" "-"))
    ]
    [rectoPageFormat (TEDIT.SINGLE.PAGEFORMAT T 25.5 3.0 (@ looks)
      (QUOTE CENTERED)
      6 6 6 6 2 NIL 12 NIL (QUOTE PICAS)
      (QUOTE (FOLIOINFO (ARABIC "Index-" "-"))
    ]
  )
)

(\BatchMethodDefs)

(METH InitIndex GetDisplayLabel (imageStream)
)
(* Specialization)

(METH InitIndex HardcopySideEffect (imageStream)
)
(* Specialization)

(METH IndexEntry Display (imageStream)
)
(* Specialization)

(METH IndexEntry GetDisplayLabel (imageStream)
```

```

NIL)

(METH IndexEntry GetHardcopyLabel (imageStream) (* Specialization)
)

(METH IndexEntry HardcopySideEffect (imageStream) (* Specialization)
)

(METH CollectIndex GetDisplayLabel (imageStream) (* Specialization)
)

(METH CollectIndex HardcopySideEffect (imageStream) (* Specialization)
)

(DEFINEQ
  (InitIndex.GetDisplayLabel
    (Method ((InitIndex GetDisplayLabel)
              self imageStream)
              (* sml "25-Dec-85 12:53")
              (* Specialization)

              "Initialize Index"))

  (InitIndex.HardcopySideEffect
    (Method ((InitIndex HardcopySideEffect)
              self imageStream)
              (* sml "25-Dec-85 12:49")
              (* Specialization)

              (STREAMPROP imageStream (QUOTE IndexTable)
                NIL)))

  (IndexEntry.Display
    (Method ((IndexEntry Display)
              self imageStream)
              (* sml "3-Jan-86 13:36")
              (* Specialization)

              [LET ((xPos (DSPXPOSITION NIL imageStream))
                    (yPos (DSPYPOSITION NIL imageStream)))
                (PROG1 (_Super)
                  (if (AND (_ self DisplayImageStream? imageStream)
                          (@ displayText?))
                      then (* Re-invert the part of the label that will be displayed in the
                           hardcopy)
                          (LET [(imageBox (_ self CachedImageBox imageStream))
                                (textWidth (STRINGWIDTH (@ text)
                                                         (_ self GetCache (QUOTE font)
                                                         (BITBLT NIL NIL NIL imageStream
                                                         (PLUS (DIFFERENCE xPos (fetch XKERN of imageBox)
                                                         (DIFFERENCE (fetch XSIZE of imageBox)
                                                         textWidth))
                                                         (DIFFERENCE yPos (fetch YDESC of imageBox))
                                                         textWidth
                                                         (fetch YSIZE of imageBox)
                                                         (QUOTE TEXTURE)
                                                         (QUOTE INVERT)
                                                         BLACKSHADE))))))]
                    (CONCAT "Index: " (@ text))))

  (IndexEntry.GetDisplayLabel
    (Method ((IndexEntry GetDisplayLabel)
              self imageStream)
              (* sml "2-Jan-86 14:06")
              (* Specialization)

              (CONCAT "Index: " (@ text))))

  (IndexEntry.GetHardcopyLabel
    (Method ((IndexEntry GetHardcopyLabel)
              self imageStream)
              (* sml "2-Jan-86 14:06")
              (* Specialization)

              (if (@ displayText?)
                  then (@ text)
                  else "")))

  (IndexEntry.HardcopySideEffect
    (Method ((IndexEntry HardcopySideEffect)
              self imageStream)
              (* sml "2-Jan-86 14:07")
              (* Specialization)

              (DECLARE (SPECVARS PAGE#))
              (LET* [(indexEntry (MKATOM (@ text)))
                    (pages (ASSOC indexEntry (STREAMPROP imageStream (QUOTE IndexTable)
                    (if (NULL pages)
                        then (* the first time this item has appeared)
                            [STREAMPROP imageStream (QUOTE IndexTable)
                              (CONS (LIST indexEntry PAGE#)
                                    (STREAMPROP imageStream (QUOTE IndexTable)
                    elseif (MEMB PAGE# (CDR pages))
                        then (* already appeared on this page)

```

```

        NIL
    else (* it appeared before, but not on this page)
        (NCONC1 pages PAGE#))))

```

(CollectIndex.GetDisplayLabel

```

(Method ((CollectIndex GetDisplayLabel)
  self imageStream)
  (* sml "25-Dec-85 12:52")
  (* Specialization)

  "Collect index"))

```

(CollectIndex.HardcopySideEffect

```

(Method ((CollectIndex HardcopySideEffect)
  self imageStream)
  (* sml "25-Dec-85 12:48")
  (* Specialization)

  [LET
    [(indexInfo (STREAMPROP imageStream (QUOTE IndexTable)
      (* get rid of the index info)
      (STREAMPROP imageStream (QUOTE IndexTable)
        NIL)
      (if indexInfo
        then
          (LET ((textStream (OPENTEXTSTREAM)))
            [for entry in [SORT indexInfo (FUNCTION (LAMBDA (x y)
              (ALPHORDER (CAR x)
                (CAR y)
                (UPPERCASEARRAY]
              do [TEDIT.INSERT textStream (MKSTRING (CAR entry))
                (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)
                  [TEDIT.INSERT textStream [CONSTANT (CONCAT (CHARACTER (CHARCODE
                    TAB]
                    (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)
                  [for pageNumber on (CDR entry)
                    do [TEDIT.INSERT textStream (MKSTRING (CAR pageNumber))
                      (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)
                        (if (NOT (NULL (CDR pageNumber)))
                          then (TEDIT.INSERT textStream ", "
                            (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)
                              (TEDIT.INSERT textStream [CONSTANT (CONCAT (CHARACTER (CHARCODE
                                EOL]
                                (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)
                              (TEDIT.LOOKS textStream (@ looks)
                                1
                                (fetch TEXTLEN of (TEXTOBJ textStream)))
                              (TEDIT.PARALOOKS textStream (@ paraLooks)
                                1
                                (fetch TEXTLEN of (TEXTOBJ textStream)))
                              [TEDIT.PAGEFORMAT textStream (TEDIT.COMPOUND.PAGEFORMAT
                                (if (LISTP (@ firstPageFormat))
                                  then (EVAL (@ firstPageFormat))
                                  else (@ firstPageFormat))
                                (if (LISTP (@ versoPageFormat))
                                  then (EVAL (@ versoPageFormat))
                                  else (@ versoPageFormat))
                                (if (LISTP (@ rectoPageFormat))
                                  then (EVAL (@ rectoPageFormat))
                                  else (@ rectoPageFormat))
                                (TEDIT textStream (CREATEW NIL "Index table"])))

```

```

)

(* * Objects that let you refer to the page number of some other item in a document)

```

```

(DEFCLASSES PageNote PageReference)

```

```

[DEFCLASS PageNote (MetaClass Class doc
  Edited:
  )
  (* Used to remember the page number, so a PageReference
  instance can find it)
  (* sml "3-Jan-86 10:40")
  (Supers EditableImageObjectMixin HardcopySideEffectObject)
  (InstanceVariables (tag NIL doc)
  (* The atomic tag used to refer to this page note.))

```

```

[DEFCLASS PageReference (MetaClass Class doc
  (* Displays the page number of a particular item. The page is determined by a PageNote instance with the same tag)
  Edited:
  )
  (* sml "3-Jan-86 10:40")
  (Supers EditableImageObjectMixin HardcopySideEffectObject)
  (InstanceVariables (tag NIL doc)
  (* The atomic name used to refer to the PageNote))

```

```

(\BatchMethodDefs)

```

```
(METH PageNote GetDisplayLabel (imageStream) (* Specialization)
```

```
)
```

```
(METH PageNote HardcopySideEffect (imageStream) (* Specialization)
```

```
)
```

```
(METH PageReference GetDisplayLabel (imageStream) (* Specialization)
```

```
)
```

```
(METH PageReference GetHardcopyLabel (imageStream) (* Specialization)
```

```
)
```

```
(DEFINEQ
```

(PageNote.GetDisplayLabel

```
(Method ((PageNote GetDisplayLabel)
  self imageStream)
```

```
(* sml "26-Dec-85 10:59")
(* Specialization)
```

```
(CONCAT "NotePage: " (@ tag)))
```

(PageNote.HardcopySideEffect

```
(Method ((PageNote HardcopySideEffect)
  self imageStream)
```

```
(* sml "13-Jan-86 16:37")
(* Specialization)
```

```
(DECLARE (SPECVARS PAGE#))
[LET* [(ts (_ self TextStream imageStream))
  (oldNotes (STREAMPROP ts (QUOTE PageNotes)
    (if oldNotes
      then (LISTPUT oldNotes (@ tag)
        PAGE#)
      else (STREAMPROP ts (QUOTE PageNotes)
        (LIST PAGE# (@ tag))))
```

(PageReference.GetDisplayLabel

```
(Method ((PageReference GetDisplayLabel)
  self imageStream)
```

```
(* sml "26-Dec-85 10:58")
(* Specialization)
```

```
(CONCAT "PageRef: " (@ tag)))
```

(PageReference.GetHardcopyLabel

```
(Method ((PageReference GetHardcopyLabel)
  self imageStream)
```

```
(* sml "27-Dec-85 15:09")
(* Specialization)
```

```
(OR (LISTGET (STREAMPROP (_ self TextStream imageStream)
  (QUOTE PageNotes))
  (@ tag))
  "nnn"))
```

```
)
```

```
(\UnbatchMethodDefs)
```

```
(* * Objects for creating an TOC with section and page numbers)
```

```
(DEFCLASSES InitTOC SectionHeading CollectTOC)
```

```
[DEFCLASS InitTOC (MetaClass Class doc
```

```
(* Initialize the section heading stuff.
Must appear before the first SectionHeading object.)
(* sml "3-Jan-86 14:01")
```

```
  Edited:
  )
```

```
(Supers EditableImageObjectMixin HardcopySideEffectObject)
```

```
(InstanceVariables (initialSectionNumbers (0)
```

```
  doc
```

```
(* the initial section numbers. Form is a list of
ChapterNumber SectionNumber SubSectionNumber ...))
```

```
]
```

```
[DEFCLASS SectionHeading (MetaClass Class Edited:
```

```
(* sml "3-Jan-86 11:02")
(* Starts a new chapter, section, sub-section, ...)
```

```
  doc
```

```
(Supers TEditableImageObjectMixin HardcopySideEffectObject)
```

```
(InstanceVariables (level 1 doc
```

```
  (displayText? T doc
```

```
(* The level of this heading))
(* should the text be included in the document?)]
```

```
[DEFCLASS CollectTOC (MetaClass Class Edited:
```

```
(* sml "3-Jan-86 10:40")
(* Collects the TOC in a document and puts it in a separate
Tedit stream)
```

```
  doc
```

```
)
```

```
(Supers EditableImageObjectMixin HardcopySideEffectObject)
```

```
(InstanceVariables (title "Table of contents" looks (FAMILY HELVETICA SIZE 12 FACE BOLD)
```

```
  paraLooks
```

```
    (QUAD CENTERED)
```

```
  doc
```

```
(* The title to center at the top of the table of contents)
```

```
)
```

```

        (looks (FAMILY HELVETICA SIZE 10)
          doc
        )
        (* the looks for the index)
        (paraLooks (TABS (NIL (72 . LEFT)
          (462 . DOTTEDRIGHT)))
          doc
        )
        (* the para looks)
        [firstPageFormat (TEDIT.SINGLE.PAGEFORMAT T 25.5 3.0 (@ looks)
          (QUOTE CENTERED)
          6 6 6 6 1 NIL NIL NIL (QUOTE PICAS)
          (QUOTE (FOLIOINFO (LOWERROMAN "-" "-"))
        [versoPageFormat (TEDIT.SINGLE.PAGEFORMAT T 25.5 3.0 (@ looks)
          (QUOTE CENTERED)
          6 6 6 6 1 NIL NIL NIL (QUOTE PICAS)
          (QUOTE (FOLIOINFO (LOWERROMAN "-" "-"))
        (rectoPageFormat (TEDIT.SINGLE.PAGEFORMAT T 25.5 3.0 (@ looks)
          (QUOTE CENTERED)
          6 6 6 6 1 NIL NIL NIL (QUOTE PICAS)
          (QUOTE (FOLIOINFO (LOWERROMAN "-" "-"))

(\BatchMethodDefs)

(METH CollectTOC GetDisplayLabel (imageStream)
  )
  (* Specialization)

(METH CollectTOC HardcopySideEffect (imageStream)
  )
  (* Specialization)

(METH InitTOC GetDisplayLabel (imageStream)
  )
  (* Specialization)

(METH InitTOC HardcopySideEffect (imageStream)
  )
  (* Specialization)

(METH SectionHeading ComputeSectionNumber (imageStream)
  )
  (* * Compute the section number for object. Return it as a string in the form "nn.nn.nn.")
  (category (SectionHeading)))

(METH SectionHeading Display (imageStream)
  )
  (* Specialization)

(METH SectionHeading GetDisplayLabel (imageStream)
  )
  (* Specialization)

(METH SectionHeading GetHardcopyLabel (imageStream)
  NIL)

(METH SectionHeading HardcopySideEffect (imageStream)
  )
  (* Specialization)

(DEFINEQ

(CollectTOC.GetDisplayLabel
  (Method ((CollectTOC GetDisplayLabel)
    self imageStream)
    )
    (* sml "25-Dec-85 12:53")
    (* Specialization)
    "Collect TOC"))

(CollectTOC.HardcopySideEffect
  (Method ((CollectTOC HardcopySideEffect)
    self imageStream)
    )
    (* sml " 2-Jan-86 13:19")
    (* Specialization)
    (STREAMPROP imageStream (QUOTE SectionHeadings)
      NIL)
    (LET [(textStream (STREAMPROP imageStream (QUOTE TOCStream)
      )
      )
      ]
      (* Format the TOC entries)
      (TEDIT.LOOKS textStream (@ looks)
        1
        (fetch TEXTLEN of (TEXTOBJ textStream)))
      (TEDIT.PARALOOKS textStream (@ paraLooks)
        1
        (fetch TEXTLEN of (TEXTOBJ textStream)))
      (* Put the title at the top)
      (TEDIT.INSERT textStream [CONCAT (@ title)
        (CONSTANT (CONCAT (CHARACTER (CHARCODE EOL))
          (CHARACTER (CHARCODE EOL))
        )
        ]
        1
        (@ title:,looks))
      (TEDIT.PARALOOKS textStream (@ title:,paraLooks)
        1 1)
      (* set up the page formatting)
      [TEDIT.PAGEFORMAT textStream (TEDIT.COMPOUND.PAGEFORMAT
        (if (LISTP (@ firstPageFormat))
          then (EVAL (@ firstPageFormat))
          else (@ firstPageFormat))

```

```

        (if (LISTP (@ versoPageFormat))
            then (EVAL (@ versoPageFormat))
            else (@ versoPageFormat))
        (if (LISTP (@ rectoPageFormat))
            then (EVAL (@ rectoPageFormat))
            else (@ rectoPageFormat])
        (TEDIT textStream (CREATEW NIL "Table of contents")))))

```

(InitTOC.GetDisplayLabel

```

(Method ((InitTOC GetDisplayLabel)
  self imageStream)
  (* sml "27-Dec-85 13:40")
  (* Specialization)

  "Initialize TOC"))

```

(InitTOC.HardcopySideEffect

```

(Method ((InitTOC HardcopySideEffect)
  self imageStream)
  (* sml "3-Jan-86 14:15")
  (* Specialization)
  (* Create the text stream to accumulate the TOC)

  (STREAMPROP imageStream (QUOTE TOCStream)
    (OPENTEXTSTREAM)) (* Initialize the section numbers)
  (STREAMPROP imageStream (QUOTE SectionHeadings)
    (REVERSE (@ initialSectionNumbers))))))

```

(SectionHeading.ComputeSectionNumber

```

(Method ((SectionHeading ComputeSectionNumber)
  self imageStream)
  (* sml "14-Nov-86 11:08")

  (* * Compute the section number for object. Return it as a string in the form "nn.nn.nn.")

```

```

  (_ self PutCache (QUOTE sectionNumber)
    (APPLY (FUNCTION CONCAT)
      (for i from 1 to (@ level) collect "n."))))

```

(SectionHeading.Display

```

(Method ((SectionHeading Display)
  self imageStream)
  (* sml "3-Jan-86 13:36")
  (* Specialization)

  [LET ((xPos (DSPXPOSITION NIL imageStream))
    (yPos (DSPYPOSITION NIL imageStream))
    (imageBox (_ self CachedImageBox imageStream)))
    (PROG1 (_Super)
      (if (_ self DisplayImageStream? imageStream)
        then (* Re-invert the part of the label that will be displayed in the
              hardcopy)
          (BITBLT NIL NIL NIL imageStream (DIFFERENCE xPos
              (fetch XKERN of imageBox))
                (DIFFERENCE yPos (fetch YDESC of imageBox))
                [if (@ displayText?)
                  then (fetch XSIZE of imageBox)
                  else (STRINGWIDTH (_ self GetCache (QUOTE sectionNumber))
                    (_ self GetCache (QUOTE font]
                    (fetch YSIZE of imageBox)
                    (QUOTE TEXTURE)
                    (QUOTE INVERT)
                    BLACKSHADE))]))))

```

(SectionHeading.GetDisplayLabel

```

(Method ((SectionHeading GetDisplayLabel)
  self imageStream)
  (* sml "14-Nov-86 11:07")
  (* Specialization)

  (if (GREATERP (@ level)
    0)
    then (CONCAT (OR (_ self GetCache (QUOTE sectionNumber))
      (_ self ComputeSectionNumber imageStream))
      " "
      (@ text))
    else (@ text))))

```

(SectionHeading.GetHardcopyLabel

```

(Method ((SectionHeading GetHardcopyLabel)
  self imageStream)
  (* sml "2-Jan-86 13:39")
  (* Specialization)

  (if (@ displayText?)
    then (CONCAT (_ self GetCache (QUOTE sectionNumber))
      " "
      (@ text))
    else (_ self GetCache (QUOTE sectionNumber))))

```

(SectionHeading.HardcopySideEffect

```

(Method ((SectionHeading HardcopySideEffect)
  self imageStream)
  (* sml "14-Nov-86 11:31"
  (* Specialization)

  (DECLARE (SPECVARS PAGE#))

  (* * Update the info cached on the stream describing the current state of the section numbers)

  (LET [(oldHeadings (STREAMPROP imageStream (QUOTE SectionHeadings)

  (* The oldHeadings is a list of the section numbers (in reverse order) of the previous SectionHeading item in the document.
  It was kindly left there by the previous SectionHeading item so we could make use of it.)

  (COND
    ((LESSP (LENGTH oldHeadings)
      (@ level))
      (for i from (ADD1 (LENGTH oldHeadings)) to (@ level)
        do (push oldHeadings 1)))
    (T (SETQ oldHeadings (NLEFT oldHeadings (@ level)))
      (add (CAR oldHeadings)
        1))) (* Be nice, and store the info for the next one)
    (STREAMPROP imageStream (QUOTE SectionHeadings)
      oldHeadings))

  (* * Stash the current section label)

  [_ self PutCache (QUOTE sectionNumber)
    (APPLY (FUNCTION CONCAT)
      (for sectionNumber in (REVERSE (STREAMPROP imageStream (QUOTE
                                                                    SectionHeadings
                                                                    )))
        join (LIST sectionNumber ".")

  (* * Stash the section label in the image stream so a SectionNote can find it)

  (STREAMPROP imageStream (QUOTE CurrentSection)
    (_ self GetCache (QUOTE sectionNumber)))

  (* * Add the info to the TOC stream. This ends up looking like -
  "nn.nn Mumble .....pageNum")

  [LET [(textStream (STREAMPROP imageStream (QUOTE TOCStream)
    [TEDIT.INSERT textStream (CONCAT (_ self GetCache (QUOTE sectionNumber))
      [CONSTANT (CONCAT (CHARACTER (CHARCODE TAB)
        (@ text))
      (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)
    [TEDIT.INSERT textStream [CONSTANT (CONCAT (CHARACTER (CHARCODE TAB)
      (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)
    [TEDIT.INSERT textStream (MKSTRING PAGE#)
      (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)
    [TEDIT.INSERT textStream [CONSTANT (CONCAT (CHARACTER (CHARCODE EOL)
      (ADD1 (fetch TEXTLEN of (TEXTOBJ textStream)))

  )

  (\UnbatchMethodDefs)

  (* * Objects that let you refer to the section number of some other item in the document)

  (DEFCLASSES SectionNote SectionReference)

  (DEFCLASS SectionNote (MetaClass Class doc
    Edited:
    )
    (Supers EditableImageObjectMixin HardcopySideEffectObject)
    (InstanceVariables (tag NIL doc
    (* Used to remember the section number, so a
    SectionReference can find it)
    (* sml "3-Jan-86 10:40")
    (* The atomic tag used to refer to this section note.])

  (DEFCLASS SectionReference (MetaClass Class doc
    (* Displays the section number of a particular item. The section number is determined by a SectionNote instance with the
    same tag.)

    Edited:
    )
    (Supers EditableImageObjectMixin HardcopySideEffectObject)
    (InstanceVariables (tag NIL doc
    (* sml "3-Jan-86 10:40")
    (* The atomic name used to refer to the SectionNote)])

  (\BatchMethodDefs)

  (METH SectionNote GetDisplayLabel (imageStream)
    NIL)

  (METH SectionNote HardcopySideEffect (imageStream)
    )
    (* Specialization)

  (METH SectionReference GetDisplayLabel (imageStream)
    )
    (* Specialization)

```

```

)

(METH SectionReference GetHardcopyLabel (imageStream) (* Specialization)
)

(DEFINEQ
  (SectionNote.GetDisplayLabel
    (Method ((SectionNote GetDisplayLabel)
              self imageStream)
              (* sml "26-Dec-85 10:59")
              (* Specialization)
              (CONCAT "NoteSection: " (@ tag))))

  (SectionNote.HardcopySideEffect
    (Method ((SectionNote HardcopySideEffect)
              self imageStream)
              (* sml "13-Jan-86 17:21")
              (* Specialization)
              [LET* [(ts (_ self TextStream imageStream))
                     (oldNotes (STREAMPROP ts (QUOTE SectionNotes)
                     (if oldNotes
                         then (LISTPUT oldNotes (@ tag)
                                      (STREAMPROP imageStream (QUOTE CurrentSection)))
                         else (STREAMPROP ts (QUOTE SectionNotes)
                                      (LIST (@ tag)
                                      (STREAMPROP imageStream (QUOTE CurrentSection)))]))

  (SectionReference.GetDisplayLabel
    (Method ((SectionReference GetDisplayLabel)
              self imageStream)
              (* sml "26-Dec-85 10:58")
              (* Specialization)
              (CONCAT "SectionRef: " (@ tag))))

  (SectionReference.GetHardcopyLabel
    (Method ((SectionReference GetHardcopyLabel)
              self imageStream)
              (* sml "13-Jan-86 16:35")
              (* Specialization)
              (OR (LISTGET (STREAMPROP (_ self TextStream imageStream)
                                      (QUOTE SectionNotes))
                    (@ tag))
                  "nn.")))

)

(\UnbatchMethodDefs)

(* * Image object for displaying pretty-printed LISP forms)

(DEFCLASSES PPIImageObject)

[DEFCLASS PPIImageObject (MetaClass Class Edited: (* sml "15-Aug-86 13:02")
  doc

  (* An object that pretty-prints a LISP form or function. Currently does not work in hardcopy mode.)

  )
  (Supers EditableImageObjectMixin LoopsImageObject)
  (InstanceVariables (form NIL function? NIL doc (* the LISP form that is PPed by the object))
    (minWidth 40 doc (* the minimum number of characters per line))
    (maxWidth 80 doc (* the maximum number of characters per line))
    (lineWidth NIL doc (* the number of characters wide in the PPed form))
    (nLines NIL doc (* the number of text lines in the PPed form))

(\BatchMethodDefs)

(METH PPIImageObject ButtonEventIn (windowStream selection relX relY window textStream button)
  (* Specialization)
  )

(METH PPIImageObject Display (imageStream) (* Specialization)
  )

(METH PPIImageObject ImageBox (imageStream currentX rightMargin) (* Specialization)
  )

(METH PPIImageObject PPForm (stream font)
  NIL)

(DEFINEQ
  (PPIImageObject.ButtonEventIn
    (Method ((PPIImageObject ButtonEventIn)
              self windowStream selection relX relY window textStream button)
              (* sml "15-Aug-86 13:02")

```


(* Specialization)

```
(SELECTQ [MENU
  (create
    MENU
    CENTERFLG _ T
    ITEMS _
    (QUOTE
      ((SaveValue (QUOTE
                    SaveValue
                    )
                  "Put this Lisp
object in
(SavedValue)"
                  (SUBITEMS
                    SaveValue
                    Inspect)))
      DEdit
      ("Edit image object"
       (QUOTE Edit])
    (DEdit (EDITL
              (LIST (@ form)))
            (QUOTE CHANGED))
    (SaveValue (_Proto
                 ($ LoopsIcon)
                 Save self)
                NIL)
    (Inspect (INSPECT self)
              NIL)
    (Edit (_ self DEditObject
             textStream)
           NIL)
  NIL)))
```

(PPIImageObject.Display

```
(Method ((PPIImageObject Display)
        self imageStream)
```

```
(* smL "15-Oct-85 18:22")
(* Specialization)
```

```

(LET [(font (FONTCREATE (QUOTE GACHA)
                        (QUOTE 10)
                        NIL 0 (IMAGESTREAMTYPE imageStream)
                        (RESETLST
                          (DSPYPOSITION [IPLUS (DSPYPOSITION NIL imageStream)
                                                (TIMES (SUB1 (@ nLines))
                                                           (FONTPROP font (QUOTE HEIGHT)
                                                                imageStream))
                          (RESETSAVE NIL (LIST (FUNCTION DSPLEFTMARGIN)
                                                (DSPLEFTMARGIN (IPLUS (DSPXPOSITION NIL imageStream)
                                                                        0.5)
                                                                imageStream)
                                                imageStream))
                          (RESETSAVE NIL (LIST (FUNCTION DSPSPACEFACTOR)
                                                (DSPSPACEFACTOR (IQUOTIENT (CHARWIDTH (CHARCODE W)
                                                                                   font)
                                                                                  (CHARWIDTH (CHARCODE SPACE)
                                                                                   font))
                                                imageStream)
                          imageStream))
      (_ self PPForm imageStream font)))]

```

(PPIImageObject.ImageBox

```
(Method ((PPIImageObject ImageBox)
        self imageStream currentX rightMargin)
```

```
(* smL " 9-Oct-85 10:30")
(* Specialization)
```

[illegible]

```

when (EQ (BIN tempFile)
        (CHARCODE EOL))
do (add nLines 1)
(CLOSEF tempFile)
(change (@ nLines)
        nLines)
(create IMAGEBOX
        XSIZE _ (TIMES (@ lineWidth)
                        (CHARWIDTH (CHARCODE W)
                                   font))
        YSIZE _ (TIMES nLines (FONTPROP font
                                       (QUOTE HEIGHT)))
        YDESC _ (FONTPROP font (QUOTE DESCENT))
        XKERN _ 0))))

```

(PPImageObject.PPForm

```

(Method ((PPImageObject PPForm)
        self stream font)

```

(* smL "14-Oct-85 13:02")

(* ** pretty-print the form to a temp file, with all sorts of special flags set)

```

(RESETLST
 (RESETSAVE **COMMENT**FLG NIL)
 (RESETSAVE PRETTYTABFLG NIL)
 (RESETSAVE NIL (LIST (FUNCTION DSPFONT)
                      (DSPFONT font stream)
                      stream))
 (RESETSAVE NIL (LIST (FUNCTION LINELENGTH)
                      (LINELENGTH (@ lineWidth)
                                   stream)
                      stream))
 (PRINTDEF (@ form)
            0
            (@ form:,function?)
            NIL NIL stream))))

```

)

(\UnbatchMethodDefs)

(* ** This is a patch -- why isn't this fn defined in the system?)

(DEFINEQ

(\DSPSPACEFACTOR.DISPLAY

```

[LAMBDA (value stream)
  1])

```

(* smL "14-Oct-85 13:16")

)

(* ** Image object for displaying a box around another image object)

(DEFCLASSES BoxedImageObject)

```

[DEFCLASS BoxedImageObject (MetaClass Class Edited:
  doc

```

(* smL " 7-Mar-86 19:03")

(* An image object that surrounds another image object with a box of variable width and shade)

```

)
(Supers EditableImageObjectMixin LoopsImageObject)
(InstanceVariables (boxedObject NIL type (Subclass EmbeddedImageObject)
  doc (* the boxed imageobject)
)
  (nestedBox NIL doc (* the size of the nested imageobject))
  (boxWidth 2 doc (* the width of the box))
  (boxWhiteSpace 1 doc (* how much of an interior white border should be used))
  (boxShade BLACKSHADE doc (* the shade to box with.))
)

```

(\BatchMethodDefs)

```

(METH BoxedImageObject ButtonEventIn (windowStream selection relX relY window textStream button)
  NIL)

```

```

(METH BoxedImageObject Display (imageStream)
  NIL)

```

```

(METH BoxedImageObject ImageBox (imageStream currentX rightMargin)
  NIL)

```

(DEFINEQ

(BoxedImageObject.ButtonEventIn

```

(Method ((BoxedImageObject ButtonEventIn)

```

```

self windowStream selection relX rely window textStream button)
(* sml "10-Mar-86 13:57")
(* Specialization)
[LET
  [ (boxedObjectRegion
    (CREATEREGION (PLUS (@ boxWidth)
                        (@ boxWhiteSpace
                          ))
      (PLUS (@ boxWidth)
            (@ boxWhiteSpace))
    (fetch XSIZE
      of (@ nestedBox))
    (fetch YSIZE
      of (@ nestedBox]
    (if (INSIDEP boxedObjectRegion relX
        rely)
      then
        (RESETLST
          (RESETSAVE
            NIL
            (LIST
              (FUNCTION
                DSPCLIPPINGREGION)
              (DSPCLIPPINGREGION
                boxedObjectRegion
                windowStream)
              windowStream))
          (RESETSAVE
            NIL
            (LIST
              (FUNCTION DSPXOFFSET)
              (DSPXOFFSET (fetch LEFT
                            of
                              boxedObjectRegion
                            )
                          windowStream)
              windowStream))
          (RESETSAVE
            NIL
            (LIST
              (FUNCTION DSPYOFFSET)
              (DSPYOFFSET (fetch BOTTOM
                              of
                                boxedObjectRegion
                              )
                          windowStream)
              windowStream))
          (_ (@ boxedObject)
            ButtonEventIn
            windowStream selection
            relX rely window
            textStream button))
      else
        (LET
          [ (menuChoice
            (MENU
              (create
                MENU
                ITEMS _
                (QUOTE ((SaveValue
                        (QUOTE SaveValue)
                        "Put this Lisp
                        object in
                        (SavedValue)"
                        (SUBITEMS SaveValue
                          Inspect))
                        ("Edit box
                        parameters"
                        (QUOTE Edit)
                        "Edit the Loops
                        object defining the
                        box"]
              (SELECTQ menuChoice
                (Edit (_ self DEditObject
                      textStream)
                  NIL)
                (SaveValue (_Proto
                          ($ LoopsIcon)
                          Save self)
                  NIL)
                (Inspect (INSPECT self)
                  NIL))
              NIL]))

```

(BoxedImageObject.Display

```

(Method ((BoxedImageObject Display)
  self imageStream)
  (* sml "14-Oct-85 13:32")
  (* Specialization)
  (LET ([boxShade (if (TEXTUREP (@ boxShade))
    then (@ boxShade)
    else (EVAL (@ boxShade)
      (shadedBoxWidth (TIMES (@ boxWidth)
        (DSPSCALE NIL imageStream)))
      (boxWidth (TIMES (PLUS (@ boxWhiteSpace)
        (@ boxWidth))
        (DSPSCALE NIL imageStream)))
      (xPosition (DSPXPOSITION NIL imageStream))
      (yPosition (DSPYPOSITION NIL imageStream)))
      (* first display the contained item)
      (DSPXPOSITION (PLUS boxWidth xPosition (fetch XKERN of (@ nestedBox)))
        imageStream)
      (DSPYPOSITION (PLUS boxWidth yPosition (fetch YDESC of (@ nestedBox)))
        imageStream)
      (_ (@ boxedObject)
        Display imageStream)
      (LET [[width (PLUS (TIMES 2 boxWidth)
        (fetch XSIZE of (@ nestedBox)
          (height (PLUS (TIMES 2 boxWidth)
            (fetch YSIZE of (@ nestedBox)
              (* draw the left edge)
              (BITBLT NIL NIL NIL imageStream xPosition yPosition shadedBoxWidth height
                (QUOTE TEXTURE)
                (QUOTE REPLACE)
                boxShade)
              (* draw the bottom edge)
              (BITBLT NIL NIL NIL imageStream xPosition yPosition width shadedBoxWidth
                (QUOTE TEXTURE)
                (QUOTE REPLACE)
                boxShade)
              (* draw the right bottom edge)
              (BITBLT NIL NIL NIL imageStream (DIFFERENCE (PLUS xPosition width)
                shadedBoxWidth)
                yPosition shadedBoxWidth height (QUOTE TEXTURE)
                (QUOTE REPLACE)
                boxShade)
              (* draw the top bottom edge)
              (BITBLT NIL NIL NIL imageStream xPosition (DIFFERENCE (PLUS yPosition
                height)
                shadedBoxWidth)
                width shadedBoxWidth (QUOTE TEXTURE)
                (QUOTE REPLACE)
                boxShade)))]])
    ]))

```

BoxedImageObject.ImageBox

```

(Method ((BoxedImageObject ImageBox)
  self imageStream currentX rightMargin)
  (* sml "8-Oct-85 19:07")
  (* Specialization)
  (LET [(boxWidth (TIMES (PLUS (@ boxWhiteSpace)
    (@ boxWidth))
    (DSPSCALE NIL imageStream)
    (change (@ nestedBox)
      (_ (@ boxedObject)
        ImageBox imageStream (PLUS currentX boxWidth)
        (DIFFERENCE rightMargin boxWidth)))
    (create IMAGEBOX
      XSIZE _ (PLUS (fetch XSIZE of (@ nestedBox))
        (TIMES 2 boxWidth))
      YSIZE _ (PLUS (fetch YSIZE of (@ nestedBox))
        (TIMES 2 boxWidth))
      YDESC _ 0
      XKERN _ 0)))]
    )
  (\UnbatchMethodDefs)

```

(* * Image objects that show the last time/place saved)

```
(DEFCLASSES WhenSavedImageObject WhenLastSaved WhereLastSaved)
```

```
[DEFCLASS WhenSavedImageObject (MetaClass AbstractClass Edited: (* sml "3-Jan-86 10:06")
  (Supers LabelImageMixin)
  (InstanceVariables (label "NotYetSaved" doc
    (labelForm NIL doc
      (* the time of last change)
      (* the form to compute the time of last change)]

```

```
[DEFCLASS WhenLastSaved (MetaClass Class Edited: (* sml "2-Jan-86 11:25")
  doc
  )
  (* The time the containing TEdit file was last put)
  (Supers WhenSavedImageObject)
  (InstanceVariables (labelForm (CONCAT (DATE)
    " by " INITIALS]

```

```
[DEFCLASS WhereLastSaved (MetaClass Class Edited: (* sml "2-Jan-86 11:25")

```

```

                                doc                                (* The name of the TEdit file where this was last put)
                                )
    (Supers WhenSavedImageObject)
    (InstanceVariables (labelForm (MKSTRING (OPENP FileStream]

(\BatchMethodDefs)

(METH WhenSavedImageObject BeforePutToFile (fileStream)           (* Specialization)
)

(METH WhenSavedImageObject Display (imageStream)
NIL)

(METH WhenSavedImageObject GetLabel (imageStream)                 (* Specialization)
)

(DEFINEQ

(WhenSavedImageObject.BeforePutToFile
(Method ((WhenSavedImageObject BeforePutToFile)
self FileStream)
(* sml "12-Dec-85 13:34")
(* Specialization)

[change (@ label)
(MKSTRING (EVAL (@ labelForm]
(_Super)))

(WhenSavedImageObject.Display
(Method ((WhenSavedImageObject Display)
self imageStream)
(* sml " 3-Jan-86 11:29")
(* Specialization)

[LET ((xPos (DSPXPOSITION NIL imageStream))
(yPos (DSPYPOSITION NIL imageStream))
(imageBox (_ self CachedImageBox imageStream)))
(PROG1 (_Super)
(if (_ self DisplayImageStream? imageStream)
then

(* Draw a box around the object, to show that it is not just plain text)

(LET ((x0 (DIFFERENCE xPos (fetch XKERN of imageBox)))
(y0 (DIFFERENCE yPos (fetch YDESC of imageBox)))
(xSize (fetch XSIZE of imageBox))
(ySize (fetch YSIZE of imageBox)))
(BITBLT NIL NIL NIL imageStream x0 y0 1 ySize (QUOTE TEXTURE
)

(QUOTE PAINT)
BLACKSHADE)
(BITBLT NIL NIL NIL imageStream (PLUS x0 (SUB1 xSize))
y0 1 ySize (QUOTE TEXTURE)
(QUOTE PAINT)
BLACKSHADE)
(BITBLT NIL NIL NIL imageStream x0 y0 xSize 1 (QUOTE TEXTURE
)

(QUOTE PAINT)
BLACKSHADE)
(BITBLT NIL NIL NIL imageStream x0 (PLUS y0 (SUB1 ySize))
xSize 1 (QUOTE TEXTURE)
(QUOTE PAINT)
BLACKSHADE))))))

(WhenSavedImageObject.GetLabel
(Method ((WhenSavedImageObject GetLabel)
self imageStream)
(* sml "22-Dec-85 21:01")
(* Specialization)

(@ label)))

)

(\UnbatchMethodDefs)

(* * Image objects or displaying text in tabular columns)

(DEFCLASSES TableTextObject)

[DEFCLASS TableTextObject (MetaClass Class Edited:
doc
)
(Supers TEditImageObject TEditableImageObjectMixin)
(InstanceVariables (widthInWs 10 doc
)
(* width of table entry; the number of Ws that will fit, in the current font)
)
(nLines NIL doc

```

```

(* number of text lines to use -
if NIL, then use as many as needed)
)
      (text "" doc (* the text to display))
      (justify left doc (* one of left, center, right))
      (verticalJustify bottom doc (* one of top, bottom, center))

(\BatchMethodDefs)

(METH TableTextObject Display (imageStream) (* Specialization)
)

(METH TableTextObject ImageBox (imageStream currentX rightMargin)
) (* Specialization)

(DEFINEQ
  (TableTextObject.Display
    (Method ((TableTextObject Display)
      self imageStream)
      (* sml "10-Jan-86 09:57")
      (* Specialization)

      [LET
        [(font (_ self GetCache (QUOTE font)))
         (imageBox (_ self CachedImageBox imageStream))
         (xPos (DSPXPOSITION NIL imageStream))
         (yPos (DSPYPOSITION NIL imageStream))
         (breakPoints (_ self GetCache (QUOTE breakPoints))
          (RESETLST
            (RESETSAVE (DSPFONT font imageStream)
              (LIST (FUNCTION DSPFONT)
                (DSPFONT NIL imageStream)
                imageStream))
            (for lineNumber from 1 to (OR (@ nLines)
              (LENGTH breakPoints))
              as breakPoint in breakPoints
              bind
                (start _ 1)
                (subString _ "nation, concived in liberty...")
                [yLinePos
                  (PLUS yPos (SELECTQ (@ verticalJustify)
                    (top (DIFFERENCE (fetch YSIZE of imageBox)
                      (FONTPROP font (QUOTE HEIGHT))))
                    (center (IQUOTIENT
                      (PLUS (TIMES (MIN (OR (@ nLines)
                        (LENGTH breakPoints))
                        (LENGTH breakPoints))
                      (FONTPROP font (QUOTE HEIGHT))))
                      (fetch YSIZE of imageBox))
                    2))
                    (bottom (TIMES (SUB1 (MIN (OR (@ nLines)
                      (LENGTH breakPoints))
                      (LENGTH breakPoints))
                    (FONTPROP font (QUOTE HEIGHT))))
                    (HELP "verticalJustify IV of self not one of (top center
                      bottom)"]
                do (SETQ subString (SUBSTRING (@ text)
                  start
                    (CAR breakPoint)
                    subString))
                  (while [MEMB (CHCON1 subString)
                    (CONSTANT (LIST (CHARCODE SP)
                      (CHARCODE TAB)
                      (CHARCODE CR)
                      (CHARCODE LF)
                      (CHARCODE EOL))
                    do (GNC subString))
                  (DSPXPOSITION (SELECTQ (@ justify)
                    (left xPos)
                    (right (PLUS xPos (DIFFERENCE (fetch XSIZE
                      of imageBox)
                    (STRINGWIDTH subString font)
                    )))
                    (center (PLUS xPos
                      (QUOTIENT (DIFFERENCE (fetch XSIZE
                      of imageBox)
                    (STRINGWIDTH subString
                      font))
                    2)))
                    (HELP "justify IV of self not one of (left center
                      right)"]
                  imageStream)
                  (DSPYPOSITION yLinePos imageStream)
                  [change yLinePos (DIFFERENCE DATUM (FONTPROP font (QUOTE HEIGHT)
                    (PRIN1 subString imageStream)

```

(SETQ start (CAR breakPoint))))))

(TableTextObject.ImageBox

```

(Method ((TableTextObject ImageBox)
  self imageStream currentX rightMargin)
  (* sml "10-Jan-86 09:58")
  (* Specialization)
  (LET [(font (_ self PutCache (QUOTE font)
    (_ self CurrentFont imageStream)
    [_ self PutCache (QUOTE breakPoints)
      (\ICONW.FORMAT.TITLE (@ text)
        font
        (TIMES (@ widthInWs)
          (STRINGWIDTH "W" font))
        (CONSTANT (LIST (CHARCODE SP)
          (CHARCODE TAB)
          (CHARCODE CR)
          (CHARCODE LF)
          (CHARCODE EOL]
      (create IMAGEBOX
        XSIZE _ (TIMES (@ widthInWs)
          (STRINGWIDTH "W" font))
        YSIZE _ (TIMES [OR (@ nLines)
          (LENGTH (_ self GetCache
            (QUOTE breakPoints]
              (FONTPROP font (QUOTE HEIGHT))))
        YDESC _ (FONTPROP font (QUOTE DESCENT))
        XKERN _ 0)))]
  )
  (\UnbatchMethodDefs)
  (PUTPROPS LOOPSIMAGEOBJECTS COPYRIGHT ("Xerox Corporation" 1985 1986))

```

FUNCTION INDEX

BoxedImageObject.ButtonEventIn	26	LoopsImageObject.CopyButtonEventIn	5
BoxedImageObject.Display	27	LoopsImageObject.Display	5
BoxedImageObject.ImageBox	28	LoopsImageObject.DisplayImageStream?	5
CollectIndex.GetDisplayLabel	19	LoopsImageObject.ImageBox	5
CollectIndex.HardcopySideEffect	19	LoopsImageObject.PrePrint	5
CollectTOC.GetDisplayLabel	21	LoopsImageObject.PrintText	5
CollectTOC.HardcopySideEffect	21	LoopsImageObject.WhenCopied	6
EditableImageObjectMixin.ButtonEventIn	13	LoopsImageObject.WhenDeleted	6
EditableImageObjectMixin.DEditObject	13	LoopsImageObject.WhenInserted	6
EditableImageObjectMixin.InstallEditSource	13	LoopsImageObject.WhenMoved	6
EditableImageObjectMixin.MakeEditSource	14	LoopsImageObject.WhenOperatedOn	6
EditableImageObjectMixin.NewInstance	14	LoopsImageObject.ButtonEventInFn	6
EnsureImageObject.GetWrappedValue	2	LoopsImageObject.CopyButtonEventInFn	7
HardcopySideEffectObject.Display	16	LoopsImageObject.CopyFn	7
HardcopySideEffectObject.GetDisplayLabel	16	LoopsImageObject.DisplayFn	7
HardcopySideEffectObject.GetHardcopyLabel	16	LoopsImageObject.GetFn	7
HardcopySideEffectObject.GetLabel	16	LoopsImageObject.ImageBoxFn	7
HardcopySideEffectObject.HardcopySideEffect	16	LoopsImageObject.PrePrintFn	7
HardcopySideEffectObject.ImageBox	17	LoopsImageObject.PutFn	7
ImageObjectWrapper.AfterGetFromFile	11	LoopsImageObject.WhenCopiedFn	7
ImageObjectWrapper.AfterPutToFile	11	LoopsImageObject.WhenDeletedFn	7
ImageObjectWrapper.ButtonEventIn	11	LoopsImageObject.WhenInsertedFn	7
ImageObjectWrapper.Copy	11	LoopsImageObject.WhenMovedFn	7
ImageObjectWrapper.CopyButtonEventIn	11	LoopsImageObject.WhenOperatedOnFn	7
ImageObjectWrapper.Display	11	PageNote.GetDisplayLabel	20
ImageObjectWrapper.GetIVValue	11	PageNote.HardcopySideEffect	20
ImageObjectWrapper.ImageBox	11	PageReference.GetDisplayLabel	20
ImageObjectWrapper.PrePrint	11	PageReference.GetHardcopyLabel	20
ImageObjectWrapper.WhenCopied	11	PPIImageObject.ButtonEventIn	24
ImageObjectWrapper.WhenDeleted	12	PPIImageObject.Display	25
ImageObjectWrapper.WhenInserted	12	PPIImageObject.ImageBox	25
ImageObjectWrapper.WhenMoved	12	PPIImageObject.PPForm	26
ImageObjectWrapper.WhenOperatedOn	12	SectionHeading.ComputeSectionNumber	22
IndexEntry.Display	18	SectionHeading.Display	22
IndexEntry.GetDisplayLabel	18	SectionHeading.GetDisplayLabel	22
IndexEntry.GetHardcopyLabel	18	SectionHeading.GetHardcopyLabel	22
IndexEntry.HardcopySideEffect	18	SectionHeading.HardcopySideEffect	22
InitIndex.GetDisplayLabel	18	SectionNote.GetDisplayLabel	24
InitIndex.HardcopySideEffect	18	SectionNote.HardcopySideEffect	24
InitTOC.GetDisplayLabel	22	SectionReference.GetDisplayLabel	24
InitTOC.HardcopySideEffect	22	SectionReference.GetHardcopyLabel	24
InsertLoopsImageObject	8	TableTextObject.Display	30
LabelImageMixin.Display	14	TableTextObject.ImageBox	31
LabelImageMixin.GetLabel	14	TEditableImageObjectMixin.ButtonEventIn	15
LabelImageMixin.ImageBox	14	TEditImageObject.AllObjects	9
LabelImageMixin.MakeEditSource	14	TEditImageObject.CurrentCharLooks	9
LIO	8	TEditImageObject.CurrentFont	9
LoopsImageObject.AfterGetFromFile	4	TEditImageObject.TEditIV	9
LoopsImageObject.AfterPutToFile	4	TEditImageObject.TextStream	10
LoopsImageObject.BeforePutToFile	4	WhenSavedImageObject.BeforePutToFile	29
LoopsImageObject.ButtonEventIn	4	WhenSavedImageObject.Display	29
LoopsImageObject.CachedImageBox	5	WhenSavedImageObject.GetLabel	29
LoopsImageObject.Copy	5	\DSPSPACEFACTOR.DISPLAY	26

VARIABLE INDEX

HardcopySideEffectObject.Visible? 17	LIOInsertCharCodes	8	LoopsImageFns	8
--------------------------------------	--------------------------	---	---------------------	---