
DOC-OBJECTS

Johannes A. G. M. Koomen
(Koomen.wbst@Xerox.com or Koomen@CS.Rochester.edu)

Uses: TEDIT, IMAGEOBJ, DATEFORMAT-EDITOR

This document last edited on October 27, 1987.

DESCRIPTION

DOC-OBJECTS is a generic, extensible interface for including image objects in TEdit documents. It hooks into TEdit by an extra entry on TEdit's middle button menu, as well as by redefining what happens on typing CTRL-O. Clicking the menu entry or typing CTRL-O brings up an *Objects* menu. Selecting an object causes an instance of the designated object to be inserted in the document at the position of the caret. Clicking outside the Objects menu has no effect. DOC-OBJECTS comes with a set of predefined Document Objects, which are described below. Additional Objects can easily be added to the Objects menu.

Predefined Objects

Time Stamp

[Document Object]

A TimeStamp reflects the date the document containing it was last PUT into a file. Each PUT causes a TimeStamp to be updated. Clicking the middle button over a TimeStamp brings up a DateFormat editor. The TimeStamp can be given any appearance consistent with the function DATEFORMAT (see IRM, Section 12.5). The object following the next colon is a TimeStamp object for this file: 15 Sep 1988 18:11 PDT (Thursday). Individual characters of a TimeStamp cannot be altered by TEdit, but a TimeStamp can be given arbitrary TEdit Looks. The DATEFORMAT-EDITOR package is automatically loaded by the DOC-OBJECTS package.

File Stamp

[Document Object]

A FileStamp reflects the name of the file into which the document containing it was last PUT. Each PUT causes a FileStamp to be updated. It cannot be edited. A FileStamp is initially displayed as "-- not yet filed --".

Include

[Document Object]

This document object is a dynamic version of the static TEdit Include command, and is intended to facilitate the unbundling of document chapters and sections, while maintaining the ability to print the entire document or any portion of it. When an Include object is created, the user is prompted for a file name. An Include object can be enabled or disabled. If it is enabled, the object shows in the TEdit window as '@Include[MySubFile.TEdit]', and the indicated file will be included during a hardcopy operation. If it is disabled, then the object shows (both in the TEdit window and on hardcopy) as '@DoNotInclude[MySubFile.TEdit]'.

Middle-clicking on an Include object pops up a menu with the following fields: "New File" (prompt for a new file name), "Edit File" (TEdit the Include file, or bring it to the top if it is already being edited), "Enable" (include the file during hardcopy), and "Disable" (do not include the file during hardcopy).

Two caveats:

- 1) For best results, make an Include object the last thing in a paragraph, or put it in a paragraph of its own, and set the line and paragraph leadings to 0. The Include object forces a paragraph break right after the Include object during hardcopy, to prevent the looks of the paragraph containing the Include object to mask the looks of the first paragraph in the file being included.
- 2) A document containing Include objects is best hardcopied from a FileBrowser window, rather than through the hardcopy command on the TEdit window menu. It will work properly either way, but it's a bit unnerving to watch TEdit trying to reflect on the display the inclusion of one or more files before hardcopy and the removal of the included files after hardcopy.

Horizontal Rule

[Document Object]

This provides a more user-friendly interface to the HRULE package (which is automatically loaded by the DOC-OBJECTS package). Upon selecting it a numberpad is brought up repeatedly, with which the user can indicate the thickness of alternating black and white lines. The resulting HRule object is inserted in the document whenever the numberpad is aborted or returns 0. The DOC-OBJECTS package also modifies the HRule object such that it can be edited: clicking the middle button over an HRule object brings up a structure editor (such as SEdit) on a list containing the thicknesses of the lines composing the HRule. This list can be altered in any way, as long as the editor returns another list of numbers (presumably of odd length).

Eval'd Form

[Document Object]

Selecting this object causes a type-in window to pop up. The value of the form typed in is assumed to be an image object. This is what TEdit used to do on typing CTRL-O. For TEdit's purpose, an image object is a Lisp value of type IMAGEOBJ, BITMAP, STRINGP, LITATOM, or REGION. The latter is assumed to refer to a region of the screen.

Screen Snap

[Document Object]

Selecting this object prompts for a region of the screen. A bitmap containing a copy of the given region of the screen is inserted in the document. This is equivalent to clicking the right button in the display's background while holding the SHIFT key down.

Extending the Document Objects interface

DocObjectsMenuCommands

[Variable]

This variable contains a list of menu items which are displayed in the Document Objects pop-up menu. It is analogous to the variable BackgroundMenuCommands (*cf.* IRM, Section 28.8). The Lisp form in each item is assumed to evaluate to an image object (as defined under Eval'd Form described above).

DocObjectsMenu

[Variable]

This variable caches the Document Objects menu. Set it to NIL whenever you alter the variable DocObjectsMenuCommands.

DocObjectsMenuFont [Variable]

This variable contains a font descriptor which is used for displaying the items in the Document Objects menu. The initial value is (FONTCREATE '(MODERN 12 BOLD)). Set the variable DocObjectsMenuFont to NIL whenever you alter DocObjectsMenuFont.

(DOCOBJ-STRING-IMAGEBOX STRING IMAGESTREAM) [Function]

A useful function for Document Objects that wish to display as a string of characters (such as a TimeStamp). The Document Object's IMAGEBOXFN can call this function to obtain an image box with the TEdit Looks that apply to the Document Object taken into account.

(DOCOBJ-WAIT-MOUSE WINDOWSTREAM) [Function]

A useful function for Document Objects that wish to assure that their buttoneventfn takes action only if the mouse buttons were let up within the Object's region (i.e., the clipping region of the Object's window stream). It returns T when the mouse buttons go up within the region, or NIL when the mouse moves out of the region while a button is still down.

Future predefined Document Objects

Watch this space for objects such as Index & Index Entry, Citation & Bibliography, ...