
Subject: Re: Another IDL twist

Posted by [Ben Tupper](#) on Fri, 17 Dec 1999 08:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

edward.s.meinel@aero.org wrote:

> Does anyone know the rhyme or reason for the calling methods for the
> built-in IDL READ_* procedures? Sometimes it is:
>

I don't know why, but I do know it drove me bananas. I wrote the following as a wrapper for most of the READ_* procedures. It is in the form of a FUNCTION rather than PROCEDURE. I've only tried it on TIFF, JPEG and GIF so far. If you try and have suggestions, please send them along.

Thanks,

Ben

```
;+
; NAME:
; ReadImage
;
; PURPOSE:
; This function returns the 2 or 3 plane image specified by the file
; keyword.
; The utility of this function is that the user doesn't need to remember
; which
; image reading procedures are actually functions. The user can
; optionally
; query the image only, or return the query structure and the image.
; The _extra
; keyword is applied each of the applicable image input procedures that
; have keywords.
;
; CATEGORY:
; Input/Output
;
; CALLING SEQUENCE:
; Result = ReadImage()
;
;
; INPUTS:
; All inputs are optional keywords. If file is not specified the user
; is prompted to selected one
;
```

```

;
;
; KEYWORD PARAMETERS:
; File: A string specifying the location of the image. If not provided,
the user
; is prompted to select a file.
; Query: Set this keyword to a named variable that, on output, will
contain the info
; structure returned from the QUERY_IMAGE function.
; JustQuery: Set this keyword to a non zero scalar to simply return the
Query structure.
; In this case, Image is returned as -1.
; Red, Green, Blue: Set these keywords to named variables that, on
output, contain
; the red, green and blue color vectors stored with the image. If the
image
; has no color vectors stored these values are undefined.
; ColorTable: Set this keyword to a named variable that, on output
contains the
; the 3,n color table array. This feature is available for JPEG images
only. If the image
; has a defined colortable stored, this value is undefined.
; _Extra: This keyword allows the user to set any of the many different
keywords
; for each image type. See documentation for each image type for
keywords
; that can be passed.
; Cancel: This keyword is set to 1 if the user cancels the
dialog_pickfile prompted, it is set to
; 0 (zero) otherwise.
;
;
; OUTPUTS:
; The ReadImage function returns the image specified by the file
keyword. If the JUSTQUERY
; keyword is set, the user cancels the Dialog_PickFile prompted, or an
error occurs then
; the function returns -1.
;
;
; COMMON BLOCKS:
; None.
;
; SIDE EFFECTS:
; None known.
;
; RESTRICTIONS:
; Requires the function ParseFileName.
;
; EXAMPLE:

```

```

; The following returns the image 'c:/images/image.tif', the red, green,
and blue
; color tables associated with the image (if any.) Also returned are
the GeoTiff structure
; (if the image is GeoTiff format) and the Info Structure returned by
the
; function QUERY_TIFF.
;
; Result = ReadImage( File='c:/images/image.tif', Red = Red, Green =
Green, Blue = Blue,$
; Query= Query, GeoTiff = GeoTiff)
;
; MODIFICATION HISTORY:
; Written by: Ben Tupper 26 SEP 99
; Pemaqid River Company
; email pemaqidriver@tidewater.net
; tel: (207) 563 - 1048
; 248 Lower Round Pond Road
; POB 106
; Bristol, ME 04539-0106
;
;
;
;-

```

```

FUNCTION ReadImage, File=File, $
Query = Query, _Extra = _Extra, $
Red=Red, Green = Green, Blue = Blue, ColorTable = ColorTable,$
JustQuery = JustQuery, cancel = cancel

```

```

Image = -1
Query = -1
Cancel = 1

```

```

If n_elements(JustQuery) EQ 0 Then JustQuery = 0
if n_elements(File) EQ 0 Then File = Dialog_PickFile()

```

```

if File EQ " Then Begin
Return, Image
EndIf

```

```

FileName = StrLowerCase(ParseFileName(File[0]))

```

```

Extension = FileName[2]

```

Case Extension of

```
'bmp': Result = Query_BMP(File, Query)
'dcm': Result = Query_DICOM(File, Query)
'gif': Result = Query_GIF(File, Query)
'jpg': Result = Query_JPEG(File, Query)
'pic': Result = Query_PICT(File, Query)
'png': Result = Query_PNG(File, Query)
'ppm': Result = Query_PPM(File, Query)
'pgm': Result = Query_PPM(File, Query)
'srf': Result = Query_SRF(File, Query)
'tif': Result = Query_TIFF(File, Query)
```

EndCase

If Result EQ 0 Then Return, Image

Cancel = 0

```
If (JustQuery NE 0) Then Begin
  Return, Image
EndIf
```

Case Extension of

```
'bmp': Image = Read_BMP(File, Red, Green, Blue, Ihdr, _Extra = _Extra)
'dcm': Image = Read_DICOM(File, Red, Green, Blue, _Extra = _Extra)
'gif': Read_GIF, File, Image, Red, Green, Blue, _Extra = _Extra
'jpg': Read_JPEG, File, Image, ColorTable, _Extra = _Extra
'pic': Read_PICT, File, Image, Red, Green, Blue
'png': Image = Read_PNG(File, Red, Green, Blue, _Extra = _Extra)
'ppm': Read_PPM, File, Image, _Extra = _Extra
'pgm': Read_PPM, File, Image, _Extra = _Extra
'srf': Read_SRF, File, Image, Red, Green, Blue
'tif': Image = Read_TIFF(File, Red, Green, Blue, _Extra = _Extra)
```

EndCase

Cancel = 0

Return, Image

End

--

Ben Tupper
Pemaquid River Company
248 Lower Round Pond Road
POB 106
Bristol, ME 04539

Tel: (207) 563-1048
Email: PemaquidRiver@tidewater.net
