
Subject: Re: findfile

Posted by [David Foster](#) on Fri, 28 May 1999 07:00:00 GMT

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

R.Bauer wrote:

>
> Hi,
>
> I got in trouble by findfile on a unix system
>
> while Windows IDL returns by findfile('C:*.*') all files
> unix (aix) IDL did not give a result if more than 3500 files in a
> directory.
> findfile('/tmp/*.*') is "
>
> If I use findfile('/tmp') I got all files.
>
> Unfortunately I have momentanly on the unix only idl5.1
>
> What is idl5.2 doing ?
>
> R.Bauer

I thought I'd post yet another version of a workaround for this.
The FILE_FIND.PRO below uses 'ls' on UNIX platforms, and you can
use the /DIR keyword to find directories, or the /RECURSIVE
keyword to make the search recursive, on UNIX systems.

I've also attached the doc file FILE_FIND.DOC.

Dave Foster

--

~~~~~  
David S. Foster      Univ. of California, San Diego  
Programmer/Analyst    Brain Image Analysis Laboratory  
foster@bial1.ucsd.edu   Department of Psychiatry  
(619) 622-5892      8950 Via La Jolla Drive, Suite 2240  
La Jolla, CA 92037  
~~~~~

```
; FILE_FIND.PRO 10-29-98 DSFoster
;
;-----
; function File_Find
;
```

```

; Takes file specification, expands it and returns a list of matching
; filenames. Returns -1 if no files are found (and argument FNAMES will
; be undefined). Returns -2 if arg FILESPEC is undefined or null.
; Returns the number of files found if no errors. Mostly borrowed from
; the PICKFILE widget.
;
; Set the DIR keyword if you want only the names of directories returned.
; In this case, FILESPEC has to specify a directory itself.
;
; Set the RECURSIVE keyword to have all files or directories searched
; recursively down the directory tree (Unix only).
;
; Modifications:
;
; 1-12-95 DSF Creation.
; 2-04-97 DSF Sort the filenames before returning. Set the keyword
; NO_SORT to disable this.
; 6-13-97 DSF Make /RECURSIVE and /DIR independent, so you can search
; recursively for files as well as directories. For non-UNIX
; platforms just call FINDFILE() (/DIR and /RECURSIVE not
; allowed).
; 10-29-98 DSF Use "find" command for /RECURSIVE to get correct results!
; Add /PATH for use with /RECURSIVE to specify directory to
; begin search. If /DIR specified use "ls" and return only
; directory names. Add "-d" argument to "ls" command for
; nonrecursive searches to avoid listing directory contents.
-----
;
```

FUNCTION file_find, filespec, fnames, DIR=dir, RECURSIVE=recursive, \$
 NO_SORT=no_sort, PATH=path

```

status = 0
fnames = ""

if (n_elements(filespec) eq 0) then begin
  status = -2
endif else if (strlen(filespec) eq 0) then begin
  status = -2
endif else begin

  if ( !version.os_family eq 'unix' ) then begin
    ON_IOERROR, io_error

    ; Use the FIND command for /RECURSIVE, and LS otherwise.

    if (keyword_set(DIR)) then begin
      if (keyword_set(RECURSIVE)) then begin
```

```

command = 'ls -IR ' + filespec + ' 2> /dev/null'
cmd = 'ls'
endif else begin
  command = 'ls -l ' + filespec + ' 2> /dev/null'
  cmd = 'ls'
endelse
endif else if (keyword_set(RECURSIVE)) then begin
  command = 'find ' + filespec + ' -type d 2> /dev/null'
  cmd = 'find'
endif else begin
  command = "ls -ld " + filespec + " 2> /dev/null"
  cmd = 'ls'
endelse

if (keyword_set(PATH)) then begin
  cd, current=curdir
  cd, path
endif

SPAWN, [/bin/sh", "-c", command], results, /NOSHELL

if (keyword_set(PATH)) then $
  cd, curdir

if (keyword_set(RESULTS)) then begin
  if (keyword_set(DIR)) then begin
    firsts = strtoupper(strmid(results, 0, 1))
    fileinds = where(firsts eq "D", found)
  endif else if (keyword_set(RECURSIVE)) then begin
    pos = strpos(results, '.')
    ind = where(pos eq 0)
    results(ind) = strmid(results(ind), 2, 1000)
    fnames = results
    found = (status = n_elements(fnames))
  endif else begin
    firsts = strtoupper(strmid(results, 0, 1))
    fileinds = where(firsts eq "F" or firsts eq "-" OR $
      firsts eq "I", found)
  endelse
  if (found GT 0) then begin
    if (cmd eq 'ls') then begin
      results = results(fileinds)
      FOR i = 0, n_elements(results) - 1 DO begin
        spaceinds = where(BYTE(results(i)) EQ 32)
        spaceindex = spaceinds(n_elements(spaceinds) - 1)
        results(i) = strmid(results(i), spaceindex + 1, 100)
      endFOR
      fnames = results
    endif
  endif
endif

```

```

if (not keyword_set(NO_SORT)) then $    ; Sort?
  fnames = fnames( sort(fnames) )
  status = n_elements(fnames)           ; Return-value
endif
endif else begin
  status = -1
endelse
endif else begin
  status = -1
endelse

endif else begin      ; Non-UNIX platforms

if (keyword_set(DIR) or keyword_set(RECURSIVE)) then begin
  message, 'Keywords DIR and RECURSIVE on UNIX platforms only'
endif else begin
  fnames = findfile(filespec, count = status)
  if (fnames(0) eq "") then $
    status = -1
  endelse

  endelse
endelse

return, status

io_error: return, -1

end

```

FILE_FIND

Takes a file specification string, probably containing wildcards, and returns a sorted array of matching filenames as the second argument. Returns the number of files found if no error.

Calling Sequence

Found = FILE_FIND(Filespec, Filenames)

Arguments

Filespec

A string representing the file-specification to match. All files matching this Filespec

will be found. Type: STRING.

Filenames

The list of filenames matching the Filespec.
Type: STRARR (an array of strings). Note
that this argument will be undefined if
an error is encountered or no files are
found. This list of filenames is NOT sorted.
(See example below.)

Keywords

DIR

Set this when you want directories returned.
Note that Filespec must specify a directory
itself. Only the directory names are returned,
not the paths.

ONLY ON UNIX SYSTEMS.

NO_SORT

The default behavior is to sort the array of
filenames to return. Set this keyword to prevent
this.

PATH

Use this with the RECURSIVE keyword to specify
the directory location where the recursive search
should begin. The default is the current directory.

RECURSIVE

Search for files having names matching Filespec
recursively in all subdirectories. Returns the
complete pathname for each file found.

ONLY ON UNIX SYSTEMS.

Outputs

Fills argument Filenames with the list of
filenames found. If an error is encountered
or no files are found, Filenames will be
undefined.

Returns the number of files found. If no files are found returns -1, and -2 if Filespec argument is undefined or null.

Example

```
found = FILE_FIND('/dir/im/*.doc', fnames)
```

```
; Now sort Fnames
```

```
fnames = fnames(SORT(fnames))
```

File Attachments

- 1) [file_find.pro](#), downloaded 92 times
 - 2) [file_find.doc](#), downloaded 88 times
-