Subject: Re: file_basename(): bug or feature?
Posted by David Fanning on Thu, 24 Feb 2005 21:28:10 GMT

View Forum Message <> Reply to Message

Benjamin Hornberger writes:

> On Unix machines, IDL's file_basename() function doesn't split a path in
> Windows format (with backslashes). On Windows machines, it splits both
> (yes, I know, on Windows you can also use both to specify a path). Is
> this a bug or a feature?
>
> Linux:
>
> IDL> filename='d:\path\to\file.ex
> IDL> print,file_basename(filename)
> d:\path\to\file.ex
> IDL> filename='d:/path/to/file.ex'
> IDL> print,file_basename(filename)
> file.ex
>
> Windows:
>
> IDL> filename='d:\path\to\file.ex'
> IDL> print,file_basename(filename)
> file.ex
> IDL> filename='d:/path/to/file.ex'
> IDL> print,file_basename(filename)
> file.ex
>
> I have a situation where a Linux machine has to handle Windows paths,
> where this is quite annoying ...
>
> Any comments?

I'm not sure it's a bug OR a feature. More likely the
sensible way to write the program, given the original
purpose. :-)

My FSC_BASE_FILENAME program has the same problem,
simply because I use PATH_SEP to obtain the path
separator. I think it would be difficult to write
a generic routine to handle all types of path separators
properly, but you could probably write a pretty good
specific one if you were sure of the types of filenames
you would be getting. I guess FSC_BASE_FILENAME is as
good a place as any to start.

  http://www.dfanning.com/programs/fsc_base_filename.pro

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/