
Subject: file paths

Posted by [David](#) on Wed, 02 Apr 1997 08:00:00 GMT

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

hi,

I want to add some code at the top of a fairly large project that will tell IDL where to look for all the subroutines it'll need. They are in various different folders, organized by their function. I want the code to compile everything automatically, even if I move the source around. (It would be messy to put all the subroutines in the same folder).

Is there any way other than help, /source to show the path of the currently executing file? If I knew the path to the executing file, I could append it (with a +) to !path to make this work, right?

Thanks in advance for any help!

cheers!

dave

David Katz
Esquimalt Defence Research Detachment
co-op student

Contact Info:

phone: 250.363.2925
fax: 250.363.2856
email (work): katz@edrd.dnd.ca
email (play): dkatz@gulf.uvic.ca
web: <http://gulf.uvic.ca/~dkatz/index.shtml>

Subject: Re: file paths

Posted by [thompson](#) on Tue, 08 Apr 1997 07:00:00 GMT

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

David <katz@edrd.dnd.ca> writes:

> hi,

> I want to add some code at the top of a fairly large project that will
> tell IDL where to look for all the subroutines it'll need. They are in
> various different folders, organized by their function. I want the code
> to compile everything automatically, even if I move the source around.
> (It would be messy to put all the subroutines in the same folder).

> Is there any way other than help, /source to show the path of the
> currently executing file? If I knew the path to the executing file, I
> could append it (with a +) to !path to make this work, right?

It seems to me that the simplest thing to do would be to organize all your folders under a single master folder. Then, by using the + prefix and expand_path to this master folder, all the subfolders (and sub-subfolders, etc.) would automatically be included. For example, in Unix one could write

```
!path = expand_path("/disk1/thompson/idl") + ":" + !path
```

and then all the subdirectories under /disk1/thompson/idl that contained .pro or .sav files would automatically be added to the path. Alternatively, one could define the IDL_PATH environment variable before calling IDL to get the same effect, e.g.

```
setenv IDL_PATH +/disk1/thompson/idl:+/usr/local/rsi/idl_4/lib:+/usr/local/rsi/idl_4/examples
```

Once all the software is in the path, you can use the routine RESOLVE_ALL to compile all routines called by a given routine, although you don't really need to do this.

Bill Thompson
